

NO-A191 073

SOFTWARE SUPPORTABILITY RISK ASSESSMENT IN OT&E  
(OPERATIONAL TEST AND EVA. (U) BDM CORP ALBUQUERQUE NM  
M HUEDNER ET AL. 31 AUG 84 BDM/A-84-496-TR

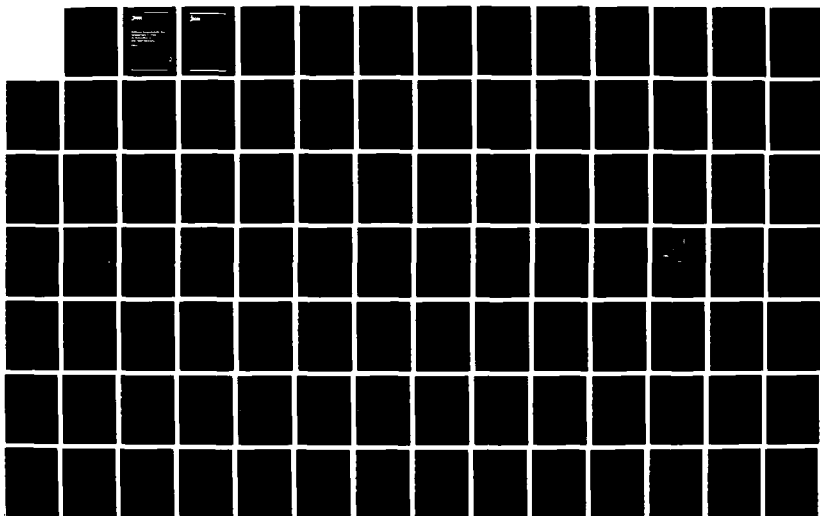
1/2

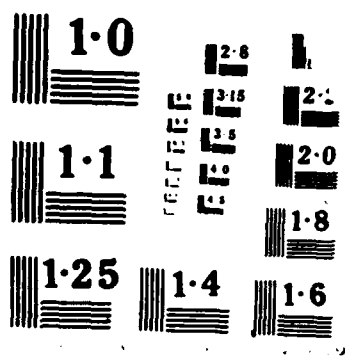
UNCLASSIFIED

F29601-00-C-0035

F/G 12/5

NL





AD-A191 873



1801 RANDOLPH ROAD, S.E. ALBUQUERQUE, NEW MEXICO 87106 (505) 848-5000 TWX 910-989-0619

# Software Supportability Risk Assessment in OT&E An Evaluation of Risk Methodologies

## FINAL

This document has been approved  
for public release and its  
distribution is unlimited.

DISTRIBUTION: UNLIMITED

DTIC  
ELECTE  
FEB 17 1988  
S E D

AUGUST 31, 1984

BDM/A-84-496-TR



1801 RANDOLPH ROAD, S.E. • ALBUQUERQUE, NEW MEXICO 87106 • (505) 848-5000

SOFTWARE SUPPORTABILITY RISK  
ASSESSMENT IN OT&E:  
AN EVALUATION OF  
RISK METHODOLOGIES

August 31, 1984

BDM/A-84-496-TR

DISTRIBUTION: UNLIMITED

Accession For	
NTIS GRA&I	<input checked="" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	



UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

AD-A191873

## REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION <b>Unclassified</b>			1b. RESTRICTIVE MARKINGS <b>None</b>		
2a. SECURITY CLASSIFICATION AUTHORITY			3. DISTRIBUTION/AVAILABILITY OF REPORT		
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE			Unlimited		
4. PERFORMING ORGANIZATION REPORT NUMBER(S)  <b>BDM/A-84-496-TR</b>			5. MONITORING ORGANIZATION REPORT NUMBER(S)		
6a. NAME OF PERFORMING ORGANIZATION  <b>The BDM Corporation</b>		6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION  <b>Air Force Operational Test and Evaluation Center/RMC</b>		
6c. ADDRESS (City, State and ZIP Code)  <b>1801 Randolph Rd., SE Albuquerque, NM 87106</b>			7b. ADDRESS (City, State and ZIP Code)  <b>Kirtland Air Force Base, NM 87117</b>		
8a. NAME OF FUNDING/SPONSORING ORGANIZATION  <b>Same as 7a</b>		8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER  <b>F29601-80-C-0035/SS304</b>		
8c. ADDRESS (City, State and ZIP Code)  <b>Same as 7b</b>			10. SOURCE OF FUNDING NOS.		
			PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.
11. TITLE (Include Security Classification)  <b>Software Supportability Risk Assessment in OT&amp;E:</b>					
12. PERSONAL AUTHOR(S) <b>W. Huebner, D. Peercy, G. Richardson</b>					
13a. TYPE OF REPORT <b>Technical</b>		13b. TIME COVERED <b>FROM 4/16/84 TO 8/31/84</b>		14. DATE OF REPORT (Yr., Mo., Day) <b>Aug 31, 1984</b>	
				15. PAGE COUNT <b>150</b>	
16. SUPPLEMENTARY NOTATION					
17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)		
FIELD	GROUP	SUB. GR.			
19. ABSTRACT (Continue on reverse if necessary and identify by block number)  <p>Assessing the software supportability risk of Air Force acquired systems is necessary to enable various decision makers to properly plan for system deployment. Risk assessment (RA) is required throughout the system acquisition life cycle. Since the perspective of OT&amp;E is focused upon the overall system mission, including supportability, methods are required which provide software testers with areas which require testing emphasis and which provide decision makers with an assessment of software and software support risk for production decisions. Due to the complexity of these requirements, it is necessary to determine the feasibility of developing and implementing a risk assessment model of software supportability with the proper system mission perspective to ultimately assist the top level decision maker.</p> <p>This report contains the results of an analysis of literature and current research to determine the level of effort and usefulness of developing and implementing a risk assessment model for software supportability (RAMSS) in OT&amp;E. SEE NEXT PAGE</p>					
20. DISTRIBUTION/AVAILABILITY OF ABSTRACT  UNCLASSIFIED/UNLIMITED <input type="checkbox"/> SAME AS RPT. <input checked="" type="checkbox"/> OTIC USERS <input type="checkbox"/>			21. ABSTRACT SECURITY CLASSIFICATION  <b>Unclassified</b>		
22a. NAME OF RESPONSIBLE INDIVIDUAL  <b>Major Gary R. Horlbeck</b>			22b. TELEPHONE NUMBER (Include Area Code) <b>505-846-1254</b>		22c. OFFICE SYMBOL  <b>LG5T</b>

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

Item 11 (cont'd):

An Evaluation of Risk Assessment Methodologies

Item 19 (cont'd):

→ This document also describes candidate RAMSS methodologies, techniques, and tools.

UNCLASSIFIED

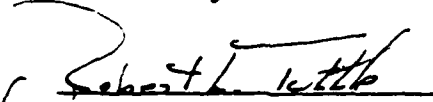
SECURITY CLASSIFICATION OF THIS PAGE

## FOREWORD

This technical report, BDM/A-84-496-TR, is submitted by The BDM Corporation, 1801 Randolph Road, S.E., Albuquerque, New Mexico, 87106, to the Air Force Operational Test and Evaluation Center, Kirtland Air Force Base, New Mexico, 87117. This report is in compliance with CDRL item A008, Contract F29601-80-C-0035, and fulfills the requirements of paragraph 7.4 of Subtask Statement 304/00, titled "Software Risk Assessment in OT&E," as amended by Subtask Statement 304/01, /02, and /03.

This report was the result of effort by Mr. Walter Huebner, Jr. (Task Leader), Dr. David Peercy, and Dr. G. Don Richardson of The BDM Corporation. The primary Subtask Statement Project Officer was Maj. Gary R. Horlbeck (AFOTEC/LG5T); the alternate Subtask Statement Project Officer was Mr. Jim Baca (AFOTEC/LG5).

Reviewed by:

  
for Fred A. Ragland  
Program Manager

## PREFACE

The use of the term "ADP" or "system" in this document is not meant to imply any particular functional category or system. In particular, the term is meant to encompass at least the four categories outlined in AFR 800-14: Category A--ADP resources in combat weapon systems and specially designed equipment; Category B--ADP resources in other systems developed under AFR 800-2; Category C--ADP resources in systems developed, acquired, and managed by AFR 80-2, AFR 65-2, AFR 71-11, and AFR 100-2; and Category D--ADP resources in general purpose, ADPS developed, acquired, and managed by the 300-series regulations and manuals. Primary application of risk assessment tools and methodologies will be to mission-critical ADP systems covered by categories A and B in accordance with AFR 800-14.



## TABLE OF CONTENTS

<u>Section</u>	<u>Page</u>
I INTRODUCTION	I-1
1.1 BACKGROUND	I-1
1.2 STUDY OBJECTIVE	I-2
1.3 STUDY APPROACH	I-3
1.4 REPORT ORGANIZATION	I-4
II EXECUTIVE SUMMARY	II-1
2.1 INTRODUCTION	II-1
2.2 ANALYSIS CONDUCTED	II-1
2.3 SCOPE OF CANDIDATES FOR AN RAMSS	II-2
2.4 LEVEL OF EFFORT FOR RAMSS DEVELOPMENT AND IMPLEMENTATION	II-3
2.5 FRAMEWORK FOR AN RAMSS	II-4
2.6 RISK ASSESSMENT METHODOLOGIES, TOOLS, TECHNIQUES	II-4
2.7 CONCLUSIONS	II-5
III RAMSS FRAMEWORK	III-1
3.1 TERMINOLOGY AND FOCUS	III-1
3.2 CRITERIA AND CONSTRAINTS	III-2
3.3 SOFTWARE SUPPORTABILITY RISK MANAGEMENT FRAMEWORK	III-6
3.3.1 The Software Supportability Risk Assessment Process	III-7
3.3.2 Evaluation Model Framework	III-17
3.3.3 Elements of Risk Management Model	III-21
3.4 MEASURES OF SOFTWARE SUPPORTABILITY RISK	III-27
3.4.1 ECS SS Profile Requirements Metrics	III-27
3.4.2 SS Evaluation Metrics	III-29
3.4.3 SS Negative Outcome Estimates	III-29
3.4.4 SS Magnitude of Consequence Estimates	III-30
3.4.5 SS Risk Levels	III-30
3.4.6 Risk Agent Acceptance Levels	III-31
3.5 REPORTING SOFTWARE SUPPORTABILITY RISK	III-31
3.6 LEVEL OF EFFORT TO DEVELOP AND IMPLEMENT A CANDIDATE RAMSS	III-33
IV RA METHODOLOGIES, TECHNIQUES, TOOLS TO SUPPORT AN RAMSS	IV-1
4.1 TERMINOLOGY AND FOCUS	IV-1
4.2 THEORETICAL FOUNDATION FOR RISK MEASUREMENT	IV-2

## TABLE OF CONTENTS (Concluded)

<u>Section</u>	<u>Page</u>
4.3 RISK METHODOLOGIES, TECHNIQUES, TOOLS	IV-6
4.3.1 Subjective Risk Techniques	IV-6
4.3.2 Objective Risk Techniques	IV-39
4.3.3 Decision Theory	IV-43
4.4 APPLICATION MODELS	IV-49
4.4.1 Georgia Tech Conceptual Model: Software Testing	IV-50
4.4.2 Proposed Fisk/Murch Model	IV-52
V REFERENCES	V-1
APPENDICES	
A ACRONYMS	A-1
B GLOSSARY OF TERMS	B-2
C POLICY DIRECTIVES	C-1

## LIST OF FIGURES

<u>Figure</u>		<u>Page</u>
3-1	Example: Software Supportability Risk Function	III-12
3-2	Example: Software Supportability Acceptance Utility Function	III-15
3-3	Evaluation Model Framework	III-18
3-4	Elements of AFOTEC Software Supportability Evaluation	III-19
3-5	SS Risk Management Model Framework	III-22
3-6	Generalized SS Risk Management Flow	III-24
3-7	Software Supportability Risk Measure Derivation	III-28
3-8	AFOTEC OT&E Reports	III-32
4-1	Theoretical Foundations of Risk	IV-3
4-2	Example: Decision Tree Techniques	IV-38
4-3	Proposed Fisk/Murch Framework for Computer Resources Risk Assessment	IV-53
4-4	Proposed Fisk/Murch Risk Assessment Analytical Procedure	IV-55
4-5	Proposed Fisk/Murch Evaluation Criteria and Risk Matrix	IV-56
4-6	Example: Proposed Fisk/Murch Model	IV-57

## LIST OF TABLES

<u>Table</u>		<u>Page</u>
2-1	RA Data Base Summary	II-2
3-1	Software Supportability Definitions	III-3
3-2	Risk Management Definitions	III-3
3-3	RAMSS Phased Development	III-34
4-1	Example: Choice Between Gambles Technique	IV-10
4-2	Example: Modified Churchman-Ackoff Technique	IV-16
4-3	Example: Delphi Procedures	IV-25
4-4	Example: Closed Form Question	IV-30

# **Section I**

## **Introduction**

## SECTION I INTRODUCTION

### 1.1 BACKGROUND.

The Air Force Operational Test and Evaluation Center (AFOTEC) has the responsibility for performing operational test and evaluation (OT&E) of assets entering the Air Force inventory. AFOTEC has developed and implemented various software OT&E methodologies. These methods have matured and have become the Air Force standard for evaluating software supportability. Each of these developed methods evaluates specific characteristics of the supportability aspects of delivered software and software support resources. These stand-alone evaluations provide AFOTEC with information to identify particular software supportability deficiencies, but do not identify overall risk associated with contractor or military ownership and organic maintenance of contractor-delivered software.

Assessing the software supportability risk of Air Force acquired systems is necessary to enable various decision makers to properly plan for system deployment. Risk assessment (RA) is required throughout the system acquisition life cycle. The perspective of OT&E is focused upon the overall system mission operation, including support. Methods are needed to provide software testers with areas which require testing emphasis, and decision makers with an assessment of the software supportability risk.

Software support for major weapon systems is becoming a major system cost factor. Major weapon systems are using more sophisticated computer systems and the support costs required for embedded software is projected to increase. Furthermore, since most enhancements to the system are dependent on software modifications, the timeliness of such software support is critical to system operational availability and effectiveness. Because of this criticality of the software support function to overall system mission operational capability, it is desired that top decision

makers be aware of the risk associated with the software supportability of a system at the conclusion of OT&E. In order to determine this risk during OT&E, AFOTEC needs to develop and implement a risk assessment model of software supportability with the proper system mission perspective to ultimately assist the top level decision maker. Due to the complexity of this requirement, it is first necessary to determine the feasibility of developing and implementing such a model.

AFOTEC produced a concept proposal (reference 5.12) for computer resources risk assessment during operational test and evaluation. This effort integrates an approach, appropriate models, and subjective and quantitative software operational and supportability measures into a management-oriented assessment of user and supporter risk. This initial involvement with the application of risk assessment to software supportability provided AFOTEC with justification to support a study of the feasibility of developing and implementing a risk assessment model for software supportability (RAMSS). The AFOTEC Subtask 304 (reference 5.0) is the statement of this feasibility study's objectives and required reports. This report documents one part of this study.

## 1.2 STUDY OBJECTIVE.

The overall objective of this task study, as stated in Subtask Statement 304/00 (reference 5.0), is to perform a feasibility study to determine the level of effort and usefulness of developing and implementing a risk assessment model for software supportability (RAMSS). The report of reference 5.31 documents the first part of the effort: to "review defense and technical literature and current research concerning methods of software supportability testing and risk assessment applicable to an OT&E environment" (reference 5.0).

The emphasis for the first part of the task was placed upon:

- a) Identifying and collecting information
  - 1) Literature search and review
  - 2) Fact-finding visits/conference
  - 3) Contact with risk assessment/software experts

- b) Assembling risk assessment data base
  - 1) Glossary of terms
  - 2) Annotated bibliography
  - 3) Key documents
  - 4) Experts/knowledgeable contacts list
  - 5) Current research list.

This report documents the second part of the overall task study: "based on the literature and research review, analyze the feasibility of developing and implementing a RAMSS that could be applied to the military systems during AFOTEC-conducted OT&E" (reference 5.0). The emphasis for the second part of the task was placed upon:

- a) Analyzing current literature and research
  - 1) DTIC, NTIS, NBS, RADC, AFOTEC, DoD, periodicals, etc.
  - 2) Potential RAMSS, or parts of RAMSS
  - 3) Continued contact with risk assessment/software experts
- b) Developing a potential framework for a feasible RAMSS
  - 1) RAMSS framework
  - 2) Risk assessment methodologies, techniques, tools.

### 1.3 STUDY APPROACH.

A three-step study approach was adopted in Subtask Statement 304/00. The steps were:

- a) Conduct a literature search and research review.
- b) Analyze the literature and research information to determine the feasibility of developing and implementing a RAMSS to be applied to military systems during AFOTEC-conducted OT&E.
- c) Identify and analyze candidate measures of supportability risk for use in developing a feasible RAMSS.

The first step results are presented in the report of reference 5.31. The literature search and review required identification of key



documents published by governmental agencies and civilian agencies. Literature searches of the Defense Technical Information Center (DTIC), National Technical Information Service (NTIS), and Rome Air Development Center (RADC) data bases were conducted. A search and review of National Bureau of Standards (NBS) publications was done. Key documents from these searches were identified and ordered for inclusion in the RA data base. Several documents from another AFOTEC subtask 294 on Computer System Security (reference 5.32) were identified. The final report bibliography will include any additional documents selected from that list. Researching the available RA technology also involved contact with a number of agencies, and identification of and discussions with RA research and evaluation personnel. The basic form and content of this data base of RA information is described in reference 5.31, but will be augmented and updated as necessary to keep the data base current throughout this study.

The second step results are presented in this report. Analysis of candidate RAMSS involved analysis of literature and research collected from step 1 in the two areas of risk assessment and software supportability. Very little crossover between the two areas was evident. Hence, it was important for the feasibility requirement of this step to analyze the elements of risk assessment, factors of software supportability, and develop a framework within which it could be determined whether these "pieces" of a RAMSS could be integrated and implemented as a RAMSS.

#### 1.4 REPORT ORGANIZATION.

The remainder of this report is organized into five sections plus a set of appendices that include the detailed information concerning the activities described in paragraph 1.3. Report sections satisfy the following objectives:

- a) Section II contains a summary of the analysis conducted, candidate RAMSSs, level of effort to develop and implement

candidate RAMSSs, a framework for a RAMSS, and potential risk assessment methodologies, techniques, and tools.

- b) Section III contains the technical details of a foundation framework within which an RAMSS could be developed and implemented. This section focuses on terminology, criteria, and constraints for such a model, a software supportability risk management framework for OT&E, a structure for the software supportability risk management process, potential measures of software supportability risk, and methods of reporting results of the risk assessment process.
- c) Section IV contains technical details on risk assessment methodologies, techniques and tools which might support development and implementation of an RAMSS. The theoretical foundation of risk assessment is briefly reviewed. Subjective and objective methodologies, techniques, and tools are described in enough detail so the nature of their applicability to an RAMSS can be better understood.
- d) Section V lists the documents whose contents have been referenced in this report.
- e) Appendix A lists acronyms used in this report.
- f) Appendix B is a glossary of terms (sources of the terms and descriptions are listed) used in this report.
- g) Appendix C is a summary of DoD and Air Force policy and directives concerning risk management.

**Section II**  
**Executive Summary**

## SECTION II EXECUTIVE SUMMARY

### 2.1 INTRODUCTION.

This section of the report provides an overview of the material presented in sections III and IV. This overview summarizes the analysis conducted, lists candidate Risk Assessment Models for Software Supportability (RAMSSs), discusses the level of effort to develop and implement candidate RAMSSs, describes an initial framework for an RAMSS, and examines potential risk assessment methodologies, techniques, and tools. Also included are the basic conclusions drawn from the study and analysis of the literature and research performed during the first phase of this subtask.

The reader is referred to appendix C of this report for a summary of material from directives of higher authorities and the military services. This material supports the need for the performance of a risk assessment study.

### 2.2 ANALYSIS CONDUCTED.

The material analyzed during this study included documents obtained from the Defense Technical Information Center (DTIC); the Rome Air Development Center (RADC); the National Technical Information Service (NTIS); Risk Analysis (RA) experts and knowledgeable personnel contacted by telephone, on fact-finding trips and at conferences; and, references in key documents. The first report (reference 5.31) of this subtask contains the list of documents and sources which were used as a data base for this study. At the time of the current report (August 31, 1984), the major sources of documents, and the document counts, are given in table 2-1. The Computer System Security (CSS) task listed below includes data from reference 5.33.

Table 2-1.

## RA Data Base Summary

<u>Source of Data</u>	<u>Quantity of Documents Identified</u>	<u>Quantity of Documents Ordered</u>
DTIC (1970-1984)	450	5
NTIS (1964-1984)	3000	53
RADC	3200	21
CSS TASK	16	16
AFOTEC	13	13
OTHER/IN HOUSE	65	65
	<u>6744</u>	<u>173</u>

Whereas a large number of documents were identified via the literature search on key words, it was found that a relatively small number of documents actually applied to the subject matter at hand. BDM personnel have obtained one-third of the total documents from other (experts, references in key documents, etc.) or in-house sources. Of the total of 173 documents, approximately one-fourth of them have been identified as "key" documents, in the sense that these documents contained information which was directly pertinent to the study of risk assessment of software supportability in the OT&E environment. These documents are listed separately in section V of this report, and form the basis for much of the analysis performed.

### 2.3 SCOPE OF CANDIDATES FOR AN RAMSS.

From the entire literature and research review, only one RAMSS has been found. This model is described by F. Fisk and W. Murch in reference 5.12, and is well known to AFOTEC. The proposed Fisk/Murch model is preliminary and is not officially sanctioned by AFOTEC. The importance of this model is its view of evaluating and reporting software user and supporter risks associated with the acceptance of computer resources, especially software. The model framework integrates aspects of current AFOTEC developed methodologies for evaluating computer resources, without restricting the possibility of including other methodologies.

One other model is currently being developed by Georgia Tech (reference 5.39). At the time of this report, the model is still in the conceptual stage. The Georgia Tech model is essentially a top down approach based upon decision theory.

Further details of these models, along with advantages and limitations, are presented in section IV of this report.

#### 2.4 LEVEL OF EFFORT FOR RAMSS DEVELOPMENT AND IMPLEMENTATION.

A major conclusion of the analysis is that it is probably feasible to develop and implement an RAMSS. Other than the proposed Fisk/Murch model and the Georgia Tech model, no current model exists. Therefore, a candidate RAMSS would have to be either one of these models or a combination of the techniques listed in section 2.6, and fall within the general RAMSS framework proposed in section III. There appear to be enough methods for risk assessment that, when the advantages and limitations of each methodology or technique are compared, either one or some combination will be selected for development and implementation. The actual comparison and selection of these techniques will be completed in the next stage of this subtask.

A preliminary examination of the development and implementation of an RAMSS indicates that the task may be better accomplished in three phases. The first phase consists of the work currently undertaken, which consists basically of a concept development. The second phase should constitute a development of model requirements, a checkout of the selected procedures, and a model design. The third phase should consist of a review of the model concept based upon information obtained from phase two, followed by the actual development and implementation of the model. The estimates of the resources required to complete these phases are:

Phase	Resources	
	Duration (Months)	Staffing (People)
I	5	3.5
II	6	3
III	8	4

See section 3.6 for more details about these preliminary estimates.

## 2.5 FRAMEWORK FOR AN RAMSS.

It is possible to develop a framework for an RAMSS which bridges the gap between the theoretical aspects of risk assessment and the application of risk to OT&E software supportability. This framework takes into account such factors as: identifying risk agents, determining negative outcomes, estimating probability of negative outcomes, reducing risk and choosing alternatives, accepting risks, and evaluating uncertainty. These factors are built into a framework based on risk determination, which is the process of identifying and estimating the magnitude of the risk, and risk evaluation, which is the complex process of determining acceptable levels of risk and alternative risk choices.

The framework factors discussed above must also be measured in order to build an effective model. The measures determined most applicable to the software supportability risk assessment process include: embedded computer system profile metrics, evaluation metrics, negative outcome probability estimates, magnitude of consequence estimates, risk levels, and risk agent acceptance levels. Some of these measures, such as the evaluation metrics, are already being captured by AFOTEC (see reference 5.1), but many are not.

Details of the proposed framework may be found in section III of this report.

## 2.6 RISK ASSESSMENT METHODOLOGIES, TOOLS, TECHNIQUES.

There are several techniques for risk assessment which have not been married into one model for an assessment of software development, but

which could be considered to assist in the development and implementation of an RAMSS. These techniques, which could also be referred to as methodologies or tools as they often are in the literature, are as follows:

- a) Choice-between-gambles technique
- b) Standard lottery
- c) Modified Churchman-Ackoff technique
- d) Delphi procedure
- e) Closed form questionnaires
- f) Bayesian analysis
- g) Network analysis
- h) Decision trees
- i) Parametric modeling
- j) Decision theory

These techniques may be grouped as subjective, objective, or some combination of both. Since it is not uncommon for entire textbooks to be devoted to any of these ten topics, it will be the intent of this report to give the reader only a flavor of the technique's mechanics, advantages, and limitations. Such a discussion is found in section IV of this document.

## 2.7 CONCLUSIONS.

The basic conclusions from the analysis of the literature and research data are:

- a) No directly applicable RAMSS exists.
- b) There is very little research in risk assessment and software where the efforts are integrated, with the possible exception in CSS, as reported in reference 5.34.
- c) The closest model to an RAMSS is the proposed Fisk/Murch model. It is feasible to implement such a model, but there are serious limitations in the theoretical foundation as discussed in section 4.4.2.



- d) The RAMSS framework presented in section III and the techniques reviewed in section IV give reasonable credibility to claim that it would be feasible to develop and implement an RAMSS.

**Section III**  
**RAMSS Framework**

### SECTION III RAMSS FRAMEWORK

#### 3.1 TERMINOLOGY AND FOCUS.

This section provides a pragmatic bridge between the theoretically-based aspects of risk assessment methodologies, techniques, and tools and the subject application area of software supportability OT&E. The theoretical foundation (set in probability theory) of risk and several risk estimation methodologies/techniques are discussed in section IV. A risk management framework and the elements of software supportability from which a feasible Risk Assessment Model for Software Supportability (RAMSS) could be developed are presented in this section. Much of this information is an expansion of material in section IV of the reference 5.31.

The key terms to understand in defining a risk assessment model for software supportability include risk, model, and software supportability. Software supportability is the subject of the assessment. Risk is what is determined by the assessment. And, the complete process as well as the descriptive characteristics of software supportability constitute an approximation to reality; that is, a model.

What the assessment process should produce is a measure of the potential that support for a specific software product (or set of products) will not satisfy requirements. This potential is represented as a probability and is "determined" for each of the possible negative outcomes (i.e., requirement or set of conditions which is defined to be unsatisfactory). Once this probability is determined for each negative outcome, thus creating a probability density function, risk can be determined by summing (or integrating) over the appropriate class of negative outcomes to arrive at the risk for that specified class. Integrating over all defined negative outcomes determines the risk for software supportability.

A model for the risk assessment process discussed above will come as close as possible to representing the real world. However, it is

important for the reader to realize that all the characteristics of software supportability which might determine whether requirements are met cannot be determined. All the possible negative outcomes cannot be determined. The potential for negative outcomes cannot be determined with exact precision. This is the nature of a model. A model has uncertainty. Thus, there is some uncertainty in the resulting risk assessment. It is important to be able to estimate the bounds for this uncertainty. This is part of a model validation and helps determine the confidence with which one can use the model. These limitations do not, however, mean that models are not beneficial, but that one must understand what is being modeled.

Some of the more important software supportability related terms are defined in table 3-1. Some of the more important risk related terms are defined in table 3-2. Other terms are defined in the Glossary (appendix B).

### 3.2 CRITERIA AND CONSTRAINTS.

Software supportability encompasses the personnel, resources, and procedures necessary to assure that software can be installed, operated, and modified to meet user requirements within acceptable limits. The OT&E of software and software support resources by the Air Force is a relatively new effort. The wide range of systems containing software, the criticality of those systems to national defense, and the ever present problem of limited OT&E resources set the broad boundaries of the general risk assessment criteria and constraints. The difference can be rather significant between the required objectives of software supportability OT&E risk assessment, and the capability of AFOTEC and other designated resources to accomplish a timely assessment of adequate depth and understanding to assist the appropriate decision makers. Therein lies the general problem statement: Is it feasible for AFOTEC with their limited resources to assess the risk of software supportability across the wide range of systems entering the Air Force inventory such that the assessment:

Table 3-1.

## Software Supportability Definitions

**SOFTWARE:**

THE PROGRAMS WHICH EXECUTE IN A COMPUTER. THE DATA INPUT, OUTPUT, CONTROLS UPON WHICH PROGRAM EXECUTION DEPENDS AND THE DOCUMENTATION WHICH DESCRIBES, IN A TEXTUAL MEDIUM, DEVELOPMENT AND MAINTENANCE OF THE PROGRAMS.

**SOFTWARE FAULT:**

THE PRESENCE OR ABSENCE OF THAT PART OF A SOFTWARE PRODUCT WHICH CAN RESULT IN SOFTWARE FAILURE.

**SOFTWARE MAINTENANCE:**

THOSE ACTIONS REQUIRED FOR:

- (1) CORRECTION. REMOVAL, CORRECTION OF SOFTWARE FAULTS
- (2) ENHANCEMENT. ADDITION/DELETION OF FEATURES FROM THE SOFTWARE
- (3) CONVLRSION. MODIFICATION OF THE SOFTWARE BECAUSE OF ENVIRONMENT (DATA HARDWARE) CHANGES

**SOFTWARE MAINTAINABILITY:**

A QUALITY OF SOFTWARE WHICH REFLECTS THE EFFORT REQUIRED TO PERFORM SOFTWARE MAINTENANCE ACTIONS.

**SOFTWARE MAINTENANCE ENVIRONMENT:**

AN INTEGRATION OF PERSONNEL SUPPORT SYSTEMS AND PHYSICAL FACILITIES FOR THE PURPOSE OF MAINTAINING SOFTWARE PRODUCTS.

**SOFTWARE MAINTENANCE MEASURES:**

MEASURES OF SOFTWARE MAINTAINABILITY AND ENVIRONMENT CAPABILITIES TO SUPPORT SOFTWARE MAINTENANCE ACTIVITY.

**SOFTWARE MANAGEMENT:**

THE POLICY, METHODOLOGY, PROCEDURES, AND GUIDELINES APPLIED IN A SOFTWARE ENVIRONMENT TO THE SOFTWARE DEVELOPMENT, MAINTENANCE ACTIVITIES. ALSO, THOSE PERSONNEL WITH SOFTWARE MANAGEMENT RESPONSIBILITIES.

**SOFTWARE SUPPORT FACILITY (SSF):**

THE FACILITY WHICH HOUSES AND PROVIDES SERVICES FOR THE SUPPORT SYSTEMS AND PERSONNEL REQUIRED TO MAINTAIN THE SOFTWARE FOR A SPECIFIC EMBEDDED COMPUTER SYSTEM.

**SOFTWARE SUPPORTABILITY:**

A MEASURE OF THE ADEQUACY OF PERSONNEL, RESOURCES, AND PROCEDURES TO FACILITATE

- (1) MODIFYING AND INSTALLING SOFTWARE
- (2) ESTABLISHING AN OPERATIONAL SOFTWARE BASELINE
- (3) MEETING USER REQUIREMENTS.

Table 3-2.

## Risk Management Definitions

**RISK IDENTIFICATION:**

THE POTENTIAL FOR REALIZATION OF UNWANTED, NEGATIVE CONSEQUENCES OF AN EVENT.

**RISK DETERMINATION:**

THE PROCESS OF IDENTIFYING AND ESTIMATING THE MAGNITUDE OF RISK.

**RISK EVALUATION:**

THE PROCESS OF DEVELOPING ACCEPTABLE LEVELS OF RISK TO INDIVIDUALS OR SOCIETY.

**RISK ASSESSMENT:**

THE TOTAL PROCESS OF QUANTIFYING A RISK AND FINDING AN ACCEPTABLE LEVEL OF THAT RISK FOR AN INDIVIDUAL, GROUP, OR SOCIETY. IT INVOLVES BOTH RISK DETERMINATION AND RISK EVALUATION.

**RISK MANAGEMENT:**

THE TOTAL PROCESS OF IDENTIFYING, CONTROLLING, AND MINIMIZING UNCERTAIN EVENTS.

- a) Has a technical depth and result format appropriate to adequately assist decision makers;
- b) Integrates at least the current AFOTEC evaluation methodologies;
- c) Has enough accuracy and repeatability to warrant confidence in its results;
- d) Is based upon a sound theoretical software and risk assessment foundation;
- e) Allows for determination of what acceptable level of risk means depending upon the identity of the risk agent and the software supportability requirements; and
- f) Is simple to use.

The AFOTEC evaluation constraints under which the above criteria must be applied include:

- a) Resource limitations
  - 1) Personnel
  - 2) Time
  - 3) Data collection (availability and accuracy)
- b) Variable environment
  - 1) Computer
  - 2) Software
  - 3) Development
  - 4) Testing/test coverage scenario
- c) Evaluation repeatability and understandability
  - 1) Evaluator experience
  - 2) Evaluation reliability
  - 3) Depth of evaluation MOEs
- d) Internal charter
  - 1) Restricts certain overlap areas (R&D)
  - 2) Early life cycle involvement not well defined

One of the major problems (reference 5.14) of software supportability is the diversity of software product and environment "forms" that any given organization must support. Software source may be written in

several different languages (even for one application system). The target operational system may have several different processors. The development environment and configuration management vary greatly across applications and are frequently not deliverable to the target maintenance organization, which is usually tasked with supporting several applications. Even when there is some early planning for software maintenance to ease such transition diversity, the "styles" of software structure and programming tend to vary within and across application systems. The DoD concept (references 5.15, 5.16) of one language (Ada) and a reasonably uniform support (development and maintenance) environment (APSE) may help lessen the diversity of future weapon systems support requirements.

The measures of software supportability are determined from the characteristics of the identified elements and actual software support activity (e.g., the measures of resources consumed during software maintenance). These measures must be reasonably accurate, easy to collect, and based upon a viable software supportability conceptual framework (or model). The scale of measurement must be consistent across the characteristics.

The model/conceptual framework of the software and its support environment, which represent the characteristics to be evaluated as part of the risk assessment process, must be simple, yet have reasonable fidelity. The framework should allow for evaluations to be conducted under varying resource constraints and test objectives (e.g., at high level or more detailed level characteristics).

The outcome of a software supportability risk assessment should be representable in a form which pinpoints high risk drivers as well as the associated detailed risk assessment and evaluation information which determines why those drivers are a high risk. It is useful if such information can be organized so that succeeding greater detail can be derived depending upon the decision maker requirements.

As an example, it should be possible to determine the overall level of the supportability risk for a delivered software system. If needed, it should also be possible to determine what level of risk is associated

with the delivered software products and the software support environment. It may be necessary to pinpoint the risk to greater levels of depth in some cases; for example, to the level of identifying which software modules are the high risk drivers or whether the support environment personnel, support systems, and/or facilities are the high risk drivers. And it should be possible to obtain risk assessment across groups of quality characteristics. For example, it may be that evaluation information indicates the software is very reliable, but is not easily modified or able to be ported to a different environment. If the user requirements during deployment of the system are likely to include any major modifications or a conversion to a new hardware system, then the risk assessment should be capable of appropriately identifying these software support risk drivers.

Risk assessment of software supportability also must be sensitive to the risk agent. The risk agent may be the developer, system user, the supporter, the evaluator, or even an indirect agent such as the general public. The perspective may vary a great deal from one agent to the next. Generally, all agents have some involvement, and if anyone has too much software support risk, even if it is only "perceived", then the other agent's risk is affected in a "real" way.

The bottom line to the decision maker will be whether the associated software supportability risk is acceptable as it relates to system performance (user) and support resource cost (supporter).

### 3.3 SOFTWARE SUPPORTABILITY RISK MANAGEMENT FRAMEWORK.

The feasibility of developing and implementing an RAMSS depends upon having an integrated framework for risk assessment and software supportability. This section provides guidance on what that framework should include. First, the process of software supportability risk assessment is examined, using AFOTEC software supportability terminology. Second, the framework for a software supportability evaluation is reviewed. Most of this framework is currently in use by AFOTEC. Some additional software support management factors are suggested. Third, elements of a



proposed risk management model are introduced, integrating the software supportability evaluation with a more generic approach to risk assessment.

This section, in combination with section IV, which describes more specific details of potential risk assessment techniques, enables the reader to see various possible approaches to developing and implementing an RAMSS. These sections do not provide details of that development and implementation since the scope of this report is limited to a feasibility analysis. Since there are no directly applicable RAMSSs and only one approach (see section 4.4.2 and reference 5.12) which combines aspects of risk assessment and software supportability, the analysis approach presented here was adopted.

### 3.3.1 The Software Supportability Risk Assessment Process.

A more structured view of risk analysis/assessment will be presented in section 3.3.2. The process described in this part includes the following aspects tailored to the software supportability terminology:

- a) Identifying risk agents
- b) Determining negative outcomes
- c) Estimating probability of negative outcome occurrences and magnitude of consequence value
- d) Reducing risk and choosing alternatives
- e) Acceptance of risk
- f) Uncertainty.

#### 3.3.1.1 Identifying Risk Agents.

Any determination of risk is relative to a particular risk agent. The identified risk agents which may be involved with an RAMSS include: supporter, user, developer, and evaluator. The proposed Fisk/Murch risk model identifies the user and supporter risk agents as primary concerns for OT&E. The developer as a risk agent could be significant if aspects of support management (as suggested in section 3.3.2) can be incorporated

into the software supportability evaluation process. It is clear that providing functional capabilities in the software during development in order to reduce supporter risk may well create a higher developer risk in delayed schedule, excessive cost, or technological stress.

#### 3.3.1.2 Determining Negative Outcomes.

Since risk is the potential for the realization of negative outcomes, these negative outcomes for software supportability should be clearly understood. Negative outcomes are relative to an identified risk agent (supporter, user, developer, evaluator). Possible negative outcomes for each of the risk agents can be determined from an embedded computer system (ECS) software maintenance profile. Negative outcomes occur when software supportability resources cannot satisfy a particular software support requirement. The software support requirements are based upon the ECS software maintenance profile. This profile specifies the expected maintenance actions which are required in order to support the mission objectives of the operational system.

An example profile might include:

- a) Types of priority support requests (e.g., emergency, urgent, normal)
- b) Expected response time for support requests (e.g., 24 hours for emergency, 1 week for urgent, 1 month for normal)
- c) Expected number (perhaps even distribution) of support requests of each priority type over a specified (e.g., 1 year) period of time (e.g., 10 emergency, 20 urgent, 100 normal)
- d) Expected number of support requests by the above categories and by type of maintenance action (correction, enhancement, conversion)
- e) Because support requests of a given type can vary greatly in complexity, it may be desirable to specify the above

information further categorized (low, med, high) by complexity with an example of each complexity category for guidance.

The support profile is defined at least for each ECS and represents a management tracking history of ECS support activity. It may also happen that a given set of resources (e.g., personnel, support systems, facilities) supports more than one ECS. In this case there will need to be a way to link the two (or more) profiles so as to reflect the risk due to this overlapping responsibility. For example, it may be very possible for resources to support an emergency request for either of two ECSs, but not both at the same time.

The support profile is a top level support activity requirements specification between any two of the risk agents. For example, a developer may be designing software products so that a particular support profile can be met. Upon evaluation (e.g., an OT&E software supportability test) the supporter may determine that the "agreed upon" support profile is violated in some way. For example, a low complexity emergency software correction may take 40 hours on the average to identify, correct, configure, and distribute, whereas the agreed upon profile requirement is 24 hours. At this point the supporter has experienced a potential negative outcome. How negative the outcome is, that is, the degree of risk, may depend upon other risk agents as well, such as the user. If the user can agree that 40 hours is acceptable, then there is no negative outcome. On the other hand, the supporter may have been less experience with the system than the developer expected, or perhaps not all the necessary support tools such as cross reference maps were available during the test. In this case, the alternative may be to reevaluate the test results under different environmental conditions. All of these alternative choices and ways to reduce the potential risk are considered as part of the risk analysis process.

### 3.3.1.3 Estimating Probability of Negative Outcome Occurrences and Magnitude of Consequence Value.

The major problems are in estimating what the magnitude of the consequence will be when a deviation from the required support profile (negative outcome) occurs, and what the probability is that any given deviation will occur. Once this process is complete for all applicable potential negative outcomes, a software supportability risk baseline has been established. Techniques for estimating these probabilities and magnitudes are discussed in section IV of this document.

A major aspect of using the above described risk baseline is to determine the relationship between the measurable software supportability factor characteristics (see section 3.3.2) and the support profile. For example, the "quality" of the source code (as evidenced by the source code listings) will have an effect upon how rapidly and effectively software maintenance actions can be accomplished. How much effect is not known. Currently AFOTEC measures the characteristics of software maintainability on a relative scale of 1 (worst) to 6 (best). A score of 2.0 may result in the inability of support resources to satisfy a required software support profile characteristic. However, it will clearly depend upon what the profile requirement is and what the support environment (SSF) capabilities are. It is useful to have some way, preferably a mathematical algorithm, of defining what the relationships among these factors are.

An extended example using current AFOTEC software supportability factors will illustrate some of the concepts of estimating risk. Suppose for a given software support facility (evaluation score of 3.8), and the associated ECS software (evaluation score of 3.0), and the required software supportability risk profile baseline (with the requirement for 24 hour turnaround for low complexity emergency maintenance correction requests), it is estimated that the probability is 0.4 that the negative outcome of 25 hour (1 hour delay) turnaround will occur. The magnitude of the consequence might be estimated as a minor inconvenience to operational readiness (during peacetime). This describes one point on a

family of related curves called probability density functions (PDFs). As each factor variable is allowed to range over possible values, or the potential negative outcome is allowed to vary, PDFs are derived. Such a probability density function (PDF) curve is created as in figure 3-1. This PDF curve is, of course, hypothesized. From this PDF, the accumulated risk across a range of possible outcomes can be determined simply by determining the area under that part of the curve (i.e., integrating over the PDF from the lower range value to the upper range value). More specific information on the theoretical foundations of risk is discussed in section 4.2 of this report.

As is easily noted, this process could become complex, especially if the number of variables is large or the data upon which probability estimation is to be done is not available. Fortunately, precise numeric values as used in the illustration above are not always needed to obtain a feel for the risk (e.g., low, medium, high). Techniques as described in section 4.3 of this report are applicable for both subjective and objective-derived data.

#### 3.3.1.4 Reducing Risk and Choosing Alternatives.

The process of reducing risk and/or choosing alternatives is part of a general risk aversion process. Knowing the potential effect due to a negative outcome will cause aversive action to be taken. Returning to the example in section 3.3.1.3 of the software supportability evaluation, one way to reduce risk is to require a developer to obtain a score of more than 3.0 as a "preventive" measure. Another possibility is to require modifications to the source code to correct the major identified deficiencies so that an overall score of 3.0 or better will be attained prior to acceptance.

Other alternatives can include upgrading related variables (e.g., the SSF) sufficient to decrease the risk. Or, perhaps the original requirement of 24 hour turnaround on emergency requests for low complexity corrective maintenance can be relaxed. The ultimate alternatives

VARIABLES:

RISK AGENT - USER

SSF SCORE - 3.8

SWM SCORE - 3.0

EMERGENCY CORRECTION

LOW COMPLEXITY REQUIREMENT - 24 HOURS

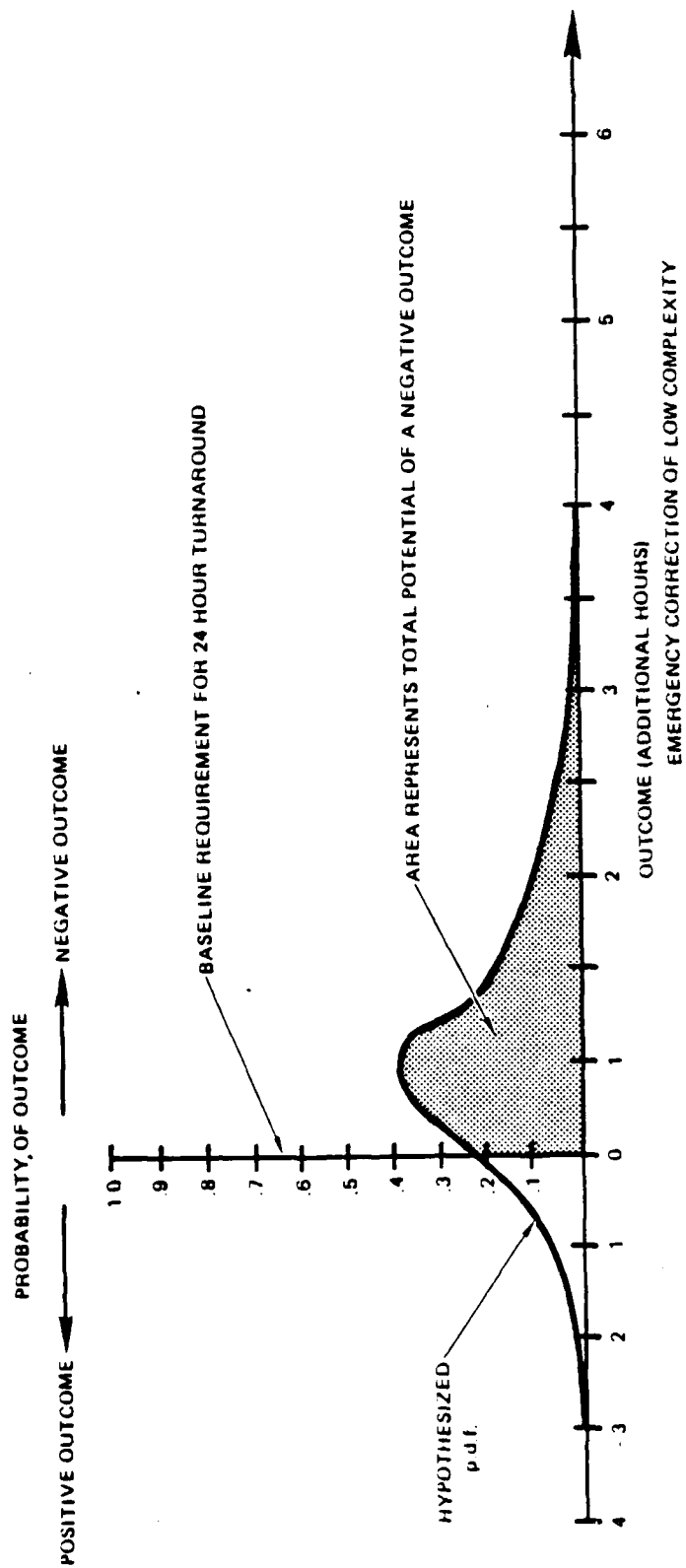


Figure 3-1. Example: Software Supportability Risk Function

are to do nothing, or to cancel any further activity on the particular subject software.

A key part of the analysis to reduce risk and/or choose alternatives is the predetermination of the cost and benefit of each possible course of action. This should be carefully (and probably simultaneously) integrated with the acceptability of the identified risk to the risk agent(s).

#### 3.3.1.5 Acceptance of Risk.

A risk agent may require an analysis of risk reduction and alternative choices before acceptability can be determined. Or, a risk agent may be able to directly specify acceptance of the risk. Once risk has been identified and the magnitude of potential negative outcomes determined, it is recommended that the risk agent be consulted as to the acceptability of the risk prior to further analysis. The analysis process itself may be costly and the potential for risk reduction small.

Risk acceptance has very broad implications. In one case an individual user may accept support risk on a particular software package. In another case the risk acceptance may involve a group of people as the risk agent and a degradation in national defense due to reduced operational effectiveness (availability) of a fighter aircraft. Risk acceptance may involve regulatory agencies, societal standards and concerns, and other political influences. Whenever there is risk it is likely that there will be more than one risk agent, but there may be a large disparity between the degree of risk which must be assumed by any one agent. This creates an atmosphere of "unfairness" which may be on the fringe of psychological perception. In any case, balancing risk among risk agents is a part of the process to reduce risk and choose alternatives and may be required as a prerequisite to risk acceptance.

Returning to the example in section 3.3.1.3, suppose the following consequences were based upon the possible negative outcome of an excessive (more than 24 hour) time to correct an emergency low complex software fault:

- a) 0-1 hours - Minor inconvenience
- b) 1-2 hours - Readiness affected
- c) 2-3 hours - Readiness impaired
- d) > 3 hours - Emergency Consequence

The level of risk acceptance depends on the probability of occurrence and the nature and magnitude of the value of the consequences that can occur. This is qualitatively illustrated in figure 3-2 for a single risk agent for our example. The abscissa shows an evenly spaced rank scale of consequence value in terms of the gross indication of the hierarchy of risk consequences arbitrarily assigned above. A logarithmic scale of probability of occurrence appears on the ordinate. Changes in the spacing in the abscissa scale will alter the specific shape of the curve, but not the general downward slope to the right hand. In other words, consequences not involving emergency consequences have higher acceptable levels of probability.

The acceptable probability of occurrence of a specific consequence value is designated as a "risk acceptance level". The profile of the acceptability of the probability of occurrence for all consequences involved in a situation is designated a "risk acceptance utility function".

Figure 3-2 illustrates two alternate risk acceptance utility functions. The top curve represents the risk acceptance utility function for a risk taker with a lower "propensity for risk acceptance" than the original risk agent. The bottom curve illustrates a higher propensity for risk acceptance; that is, the risk taker is more likely to "take a chance" than the original risk agent.

More details concerning risk acceptance and estimating risk acceptance can be found in Rowe's book (reference 5.25) and several other references. Some of the estimating techniques discussed in section IV of this report are applicable.

#### 3.3.1.6 Uncertainty.

With every descriptive process there is uncertainty. With every measurement process there is uncertainty. In order for AFOTEC to develop



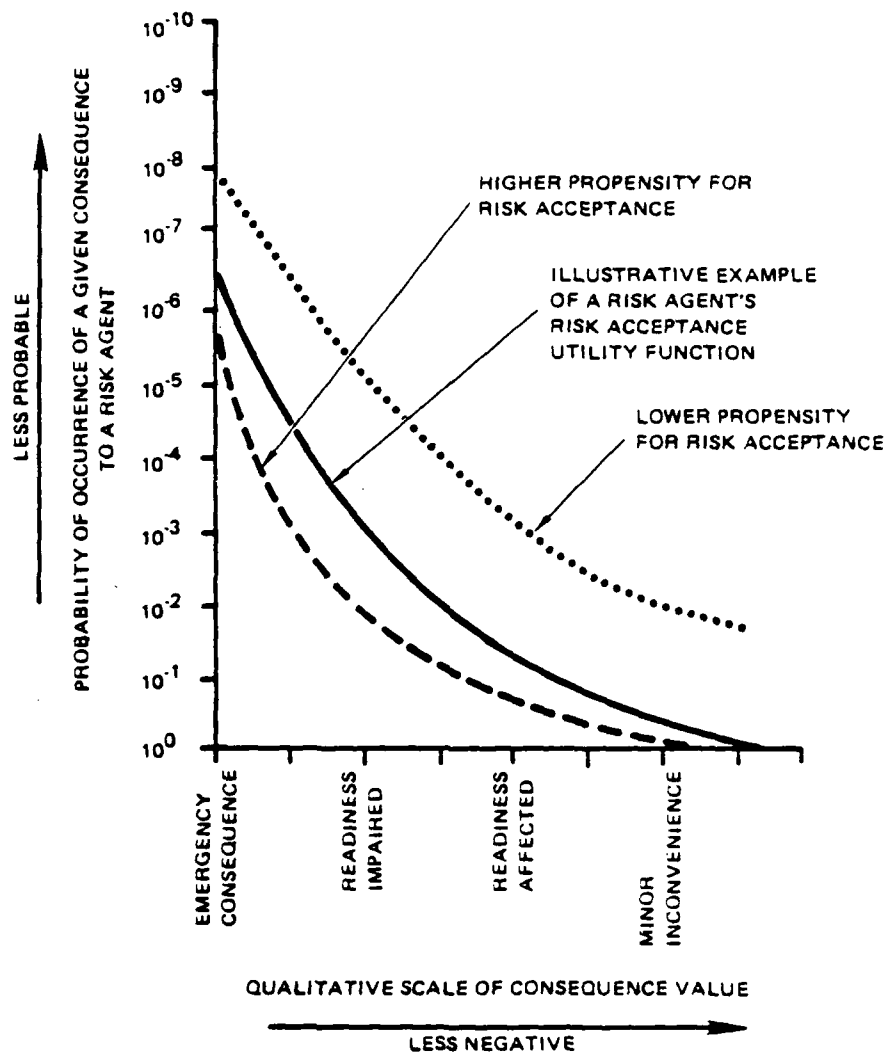


Figure 3-2. Example: Software Supportability Acceptance Utility Function

an RAMSS. it will be necessary to understand and control the uncertainty in the risk assessment process.

It is not possible to completely identify all the software supportability factors. Thus, a model is created as an approximation to the reality. Currently, this model is a hierarchy of increasingly more specific characteristics of software supportability (see section 3.3.2 and reference 5.1 for more details). This disparity between the "real" description and the "model" description of software supportability is termed descriptive uncertainty.

With each characteristic in the descriptive model hierarchy there is a corresponding evaluation measurement scale. Scales for measurement include:

- a) Nominal Scale (Identify-Taxonomy). A classification of items that can be distinguished from one another by one or more properties.
- b) Ordinal Scale (Order-Rank). An ordering (ranking) of items by the degree they obtain some criterion.
- c) Cardinal Scale (Interval). A continuous scale between two end points, neither of which is necessarily fixed.
- d) Ratio Scale (Zero reference). A cardinal scale with one end point fixed by reference to an absolute physical end point (e.g., absolute zero on the Kelvin temperature scale), from which are developed other cardinal scales, all of which are related by simple ratios.

The proposed Fisk/Murch model (reference 5.12) is based upon a six point scale of measurement: F -completely disagree; E; D; C; B; A -completely agree, which can also be considered to have integer values from 1 to 6. This scale is a cardinal scale. There is uncertainty in the scale and with the scale values assigned to software supportability characteristics as part of an evaluation. This is called measurement uncertainty. As values with measurement uncertainty are aggregated to obtain "higher level" information, the measurement uncertainty is also aggregated. This is the "compound error" problem.

One of the major requirements of an RAMSS for AFOTEC is to properly balance the descriptive uncertainty due to lack of model fidelity with the measurement uncertainty of evaluating model characteristics, estimating potential for negative software supportability outcomes, and assessing the magnitude of the consequence and acceptability level to the user and supporter of the software.

### 3.3.2 Evaluation Model Framework.

An RAMSS is by definition a model of real software supportability risk. The fidelity of such a model will depend upon the defined evaluation factors of software supportability, factor criteria and lower level characteristics, and the capability of the model to relate measures of these characteristics to software supportability risk baselines. An evaluation model framework which currently follows AFOTEC implemented evaluation methodologies is shown in figure 3-3. This figure illustrates the modeling of software supportability through a hierarchy of factors, criteria, and characteristics in succeeding more detailed descriptive levels. At each level, a measurement scale is applied to the evaluation of the information at that level. Note that as the descriptive certainty of the information increases (i.e., one proceeds to lower levels of detail in the hierarchy), the uncertainty in accumulated measurement accuracy (combination of lower level measures into higher level measures) also increases. These inversely desirable results mean that the risk assessment level, i.e., the lowest hierarchy level on which risk assessment is based, must be carefully determined so that overall uncertainty in the assessment results is minimized.

The current AFOTEC software supportability factors plus some recommended new factors (dotted lines) are shown in figure 3-4. All elements of figure 3-4 except those in dotted boxes are explained in detail in references 5.1 and 5.12. Only the elements in dotted boxes will be briefly described.

The software support management evaluation is necessary to allow for variance in software support risk which may be more directly a management

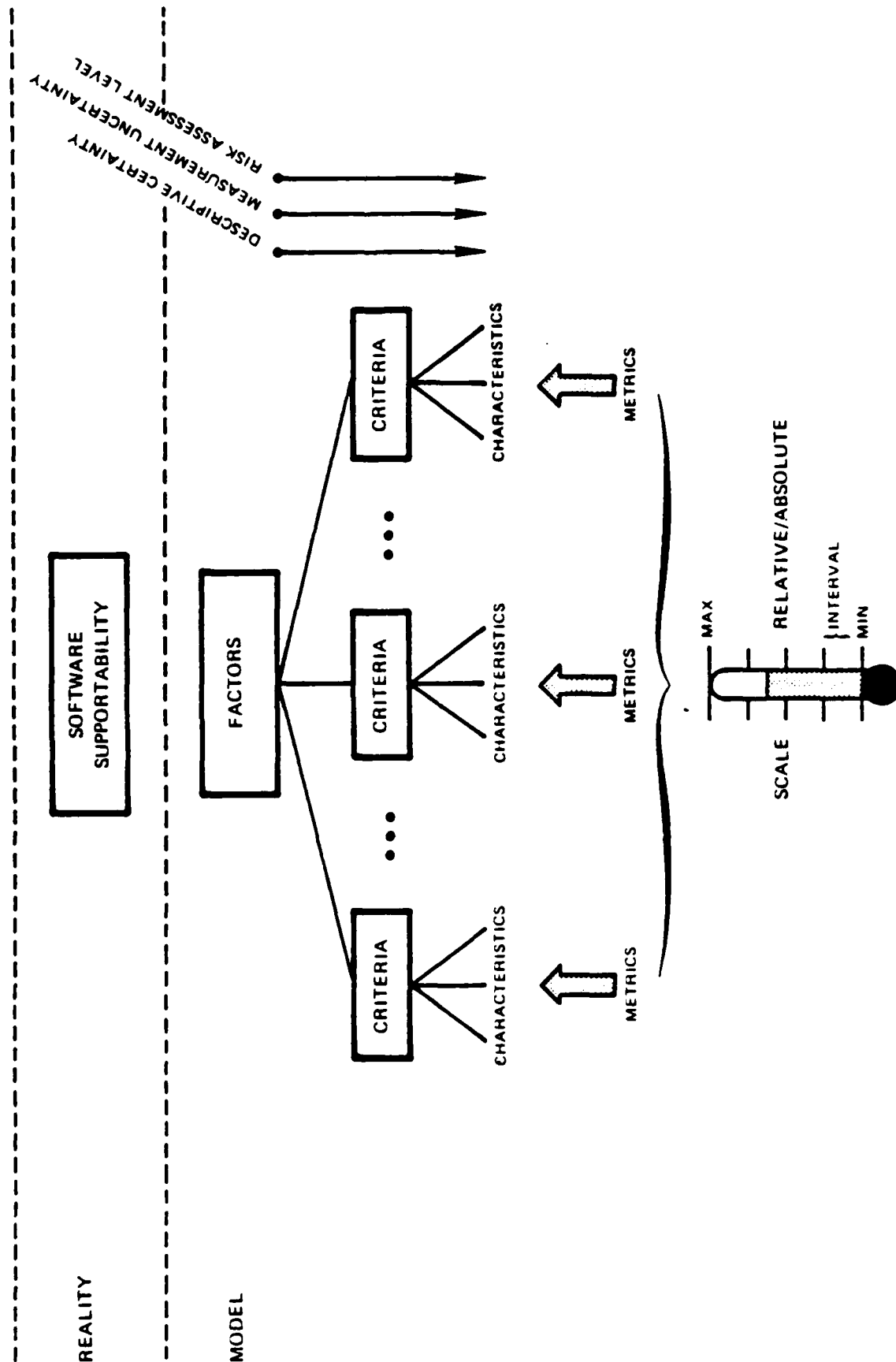


Figure 3-3. Evaluation Model Framework

## THE BDM CORPORATION

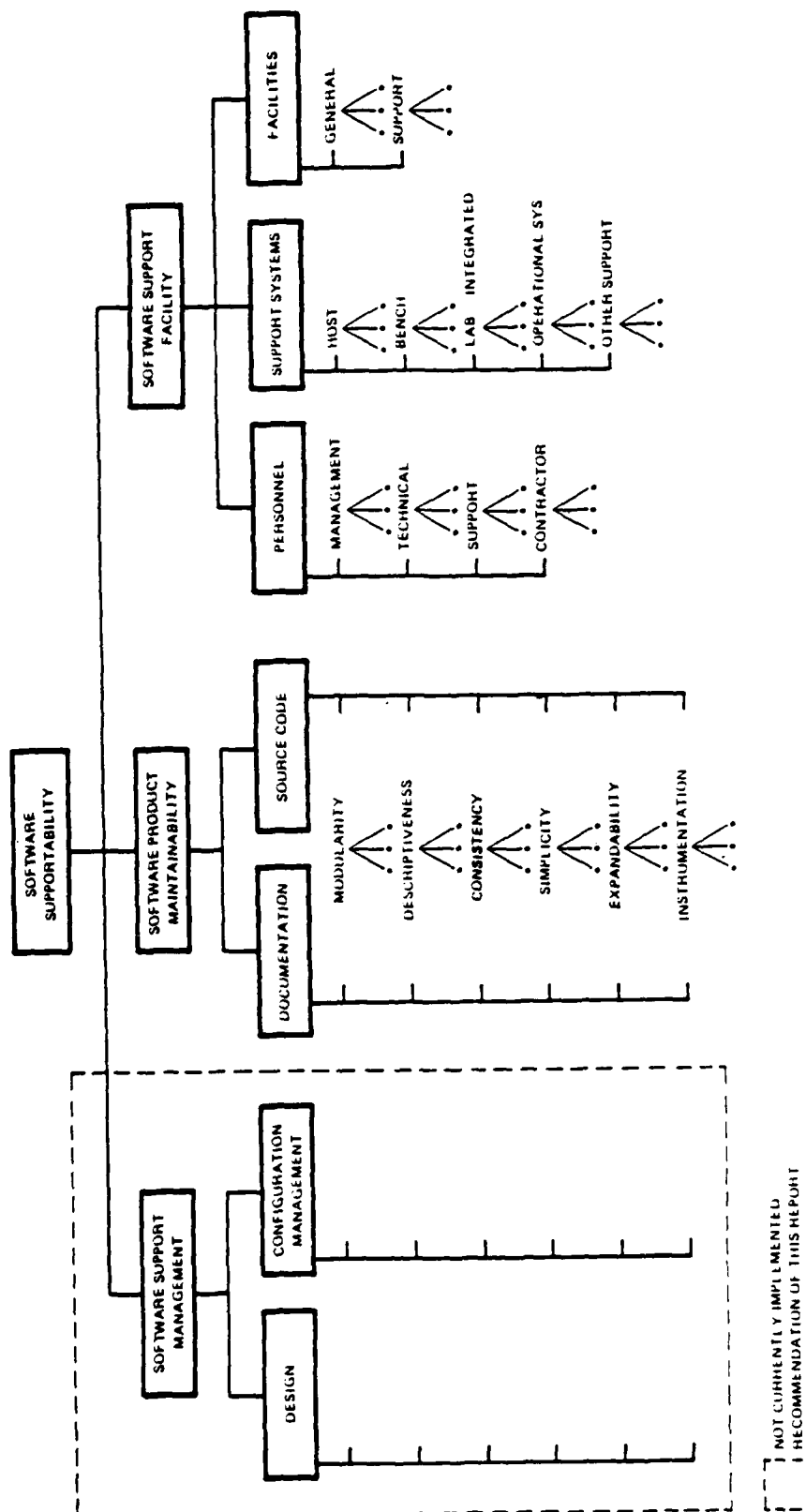


Figure 3-4. Elements of AFOTEC Software Supportability Evaluation

function than can be identified by either software product quality or the software support capabilities. This evaluation has at least two test factors: design and configuration management.

One of the important links to software supportability OT&E is the software system design which is currently only evaluated in a limited way (and then only through formal specifications). This design for software supportability is captured in the software products, the Operational/Support Configuration Management Procedures, the various test plans, the Computer Resources Integrated Support Plan, and by various development management techniques (e.g., chief programmer team, rapid prototyping) and acquisition management techniques (e.g., risk monitoring through Data Item Description metric requirements, IV&V risk assessment). Design of a system for the mission objective and software supportability is critical to reducing support risks (e.g., excessive support costs, degraded turn-around).

An OT&E software supportability design evaluation can be done early in the full scale development (or even demonstration and validation) acquisition phase. It would provide early guidance to AFOTEC for adopting a test and evaluation strategy during OT&E. This strategy could thus stress identified risk drivers (such as particular functional subsystems or modules) and help optimize the application of limited evaluation resources to maximize risk identification. This would lead to a reduced uncertainty in the residual risk eventually identified through the adopted test strategy during OT&E at the production and deployment acceptance decision point.

There are two major processes which characterize the software support function. One is the software maintenance process. The other process is software configuration management. The essence of most of the AFOTEC SSF evaluation is in measuring the capability of the SSF to support the software maintenance process within the bounds of the ECS maintenance profile.

Part of the SSF evaluation capability is to specify an "other support system" (reference figure 3-4) as a specific configuration

management system. Some of the technical characteristics of configuration management can be evaluated in this manner to minimal depth. However, a major part of configuration management is management, i.e., procedures, administration, control boards, and organizational interfaces. The AFOTEC SSF evaluation does not capture these management aspects. This "structure" begins to form very early in the system acquisition life cycle. Frequently project management places emphasis upon development configuration management with little attention paid to the transition from development to support or the support configuration management. OT&E needs to be aware of early acquisition decisions and risk evaluations concerning the software configuration management plan and how that plan specifies the transition and follow on support concepts. The Computer Resources Integrated Support Plan and the Operational/Support Configuration Management Procedures should highlight major aspects of what those requirements need to be.

### 3.3.3 Elements of Risk Management Model.

A software supportability (SS) risk management model incorporates: the software supportability test and evaluation; the risk assessment/analysis framework as applied to software supportability concerns (the RAMSS); the software supportability risk management function conducted by support MAJCOMs or Special Operating Agencies; and the appropriate system risk management by organizational agencies (e.g., the Designated Approving Authority and other agencies are interested in risk assessment for concerns in addition to software susceptibility). The basic elements of such a model are illustrated in figure 3-5.

The model illustrates the interrelationships among management (e.g., the Designated Approving Authority at the mission system management level, and program support management at the ECS level), the technical aspects of measuring software supportability through test and evaluation, and the analysis of potential negative and positive outcomes for software supportability impact and risk. A generalized process flow which represents a typical sequence of risk management activity within the model

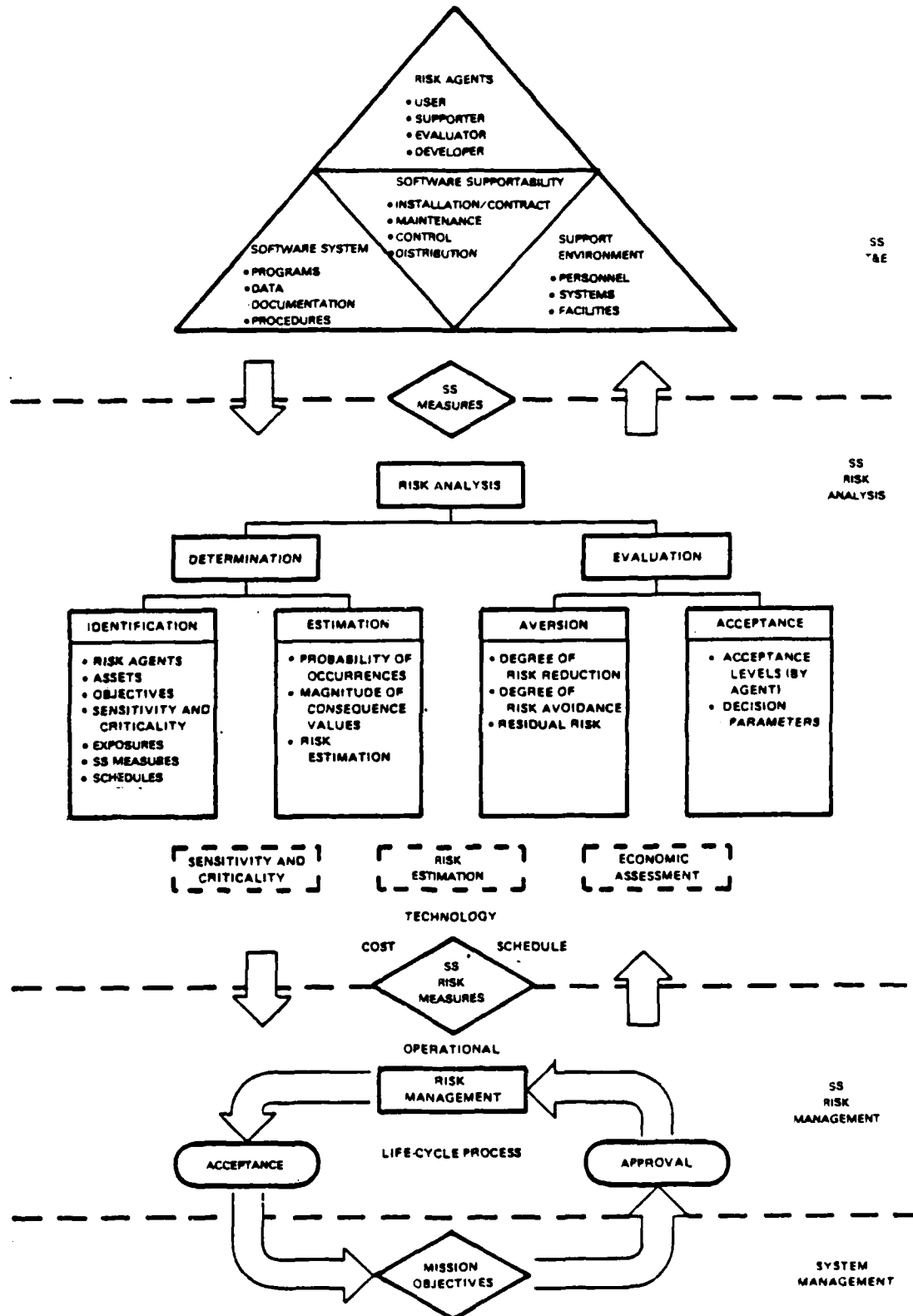


Figure 3-5. SS Risk Management Model Framework



framework is shown in figure 3-6. The following paragraphs will discuss figure 3-5. the model framework, from bottom to top.

#### 3.3.3.1 System Management.

System management controls the overall mission objectives and the allocation of those objectives to system requirements. The results of system acceptance tests are reviewed against mission objectives prior to system approval. This function is primarily allocated to the system Designated Approving Authority, assisted by appropriate Air Force agencies, MAJCOMS, Special Operating Agencies, and so forth, as necessary.

#### 3.3.3.2 SS Risk Management.

SS Risk Management includes coordination of risk analysis functions, integration with SS OT&E. and approval throughout the system life cycle. Acceptance is the result of software specification requirements being met through technical SS test and evaluations. SS risk management is responsible for determining whether the uncertainty in these results as described through the risk analysis process is acceptable for formal acceptance test requirements. The basic risk parameters as described in terms of cost, schedule, and technology impact are major inputs to this decision process. SS risk management also passes the overall software system objectives to the risk analysis process for consideration. And, SS risk management presents the results of any risk analysis as necessary to system management prior to any final approval decisions.

#### 3.3.3.3 SS Risk Analysis.

The SS Risk Analysis framework is based upon risk determination: the process of identifying and estimating the magnitude of risk; and, risk evaluation: the complex process of determining acceptable levels of risk and alternative risk choices.

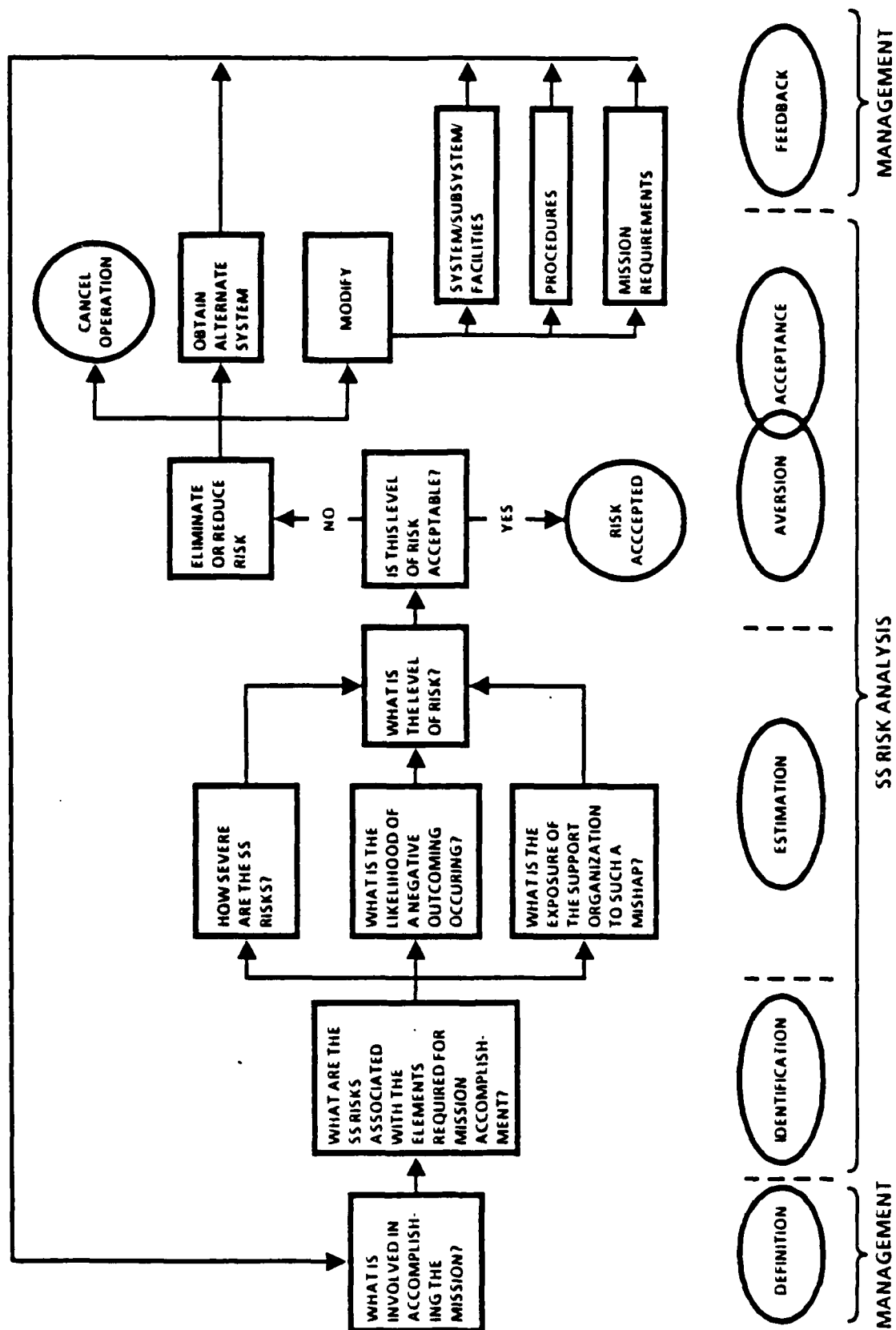


Figure 3-6. Generalized SS Risk Management Flow

- a) Identification. Identification includes determination of the risk agents, risk analysis team members, SS assets, overall SS objectives, potential negative events, possible SS MOEs, required schedules and deliverables, and the system sensitivity and process criticality. Identification also includes recognition of any changes or new relationships in any of the above risk parameters.
- b) Estimation. Estimation involves determination (either quantitatively or qualitatively) of the likelihood of an identified risk occurring, the frequency of occurrence, and the criticality or consequence if it does occur. These estimates are dependent upon the risk agent (e.g., DAA, MAJCOM/SOA, OPR, developer, using/supporting command) and are represented on scales commensurate with the required risk analysis detail as identified in a).
- c) Aversion. Aversion involves the evaluation of how the estimated level of risk might be reduced through alternatives and avoidance, and by what degree. Again, the scale of measurement might be (low, medium, high) or a detailed probability. Residual risk is determined after all alternatives and avoidance techniques have reduced the risk as much as possible.
- d) Acceptance. Acceptance represents the evaluated willingness of the particular risk agent to accept a specific level of risk (the residual risk) to obtain some gain or benefit. Part of this process is to properly represent all risk decision parameters so that acceptance can be reasonably ascertained directly from the parameters.

Risk estimation is an essential part of both the estimation and aversion functions. Potential qualitative, quantitative, and hybrid techniques which could be used are described in section IV of this document. Economic assessment is part of both the aversion and acceptance functions. The economic assessment provides the means for determining

the feasibility and relative value of the alternate software supportability risk reduction measures. A somewhat broader interpretation of "economic" to mean not just a dollar cost, but time, labor and perhaps other resource cost is very useful since it may not be possible to reduce a software supportability residual risk to a dollar cost. More detailed information concerning the risk analysis aspects of this model can be found in reference 5.25.

#### 3.3.3.4 SS OT&E.

The elements of an SS OT&E are briefly described in section 3.3.7. The SS OT&E process is described in reference 5.1. The importance of SS OT&E to risk management is in the derivation of SS evaluation measures for use in risk analysis. In addition, there is significant interdependence among SS OT&E; risk identification of potential negative events and SS risk reduction measures; and risk aversion analysis of alternatives and degree of risk reduction. During much of the process, the distinction between risk analysis and SS OT&E may be imperceptible, but the two areas do have reasonably distinct general objectives.

#### 3.3.3.5 Model Characteristics.

At any level of the model, iteration with adjacent levels is very likely. For example, identification of SS assets, potential exposure to SS risks, SS measures, and so forth derives partially from system mission objectives and partially from a technology assessment of SS OT&E results. Once the risk identification function has been "completed", SS OT&E will determine values of SS measures which can be used to integrate into the risk estimation and aversion process. But, this process may uncover other potential risk exposures previously identified, but which should be considered. Thus, the risk identification function is "reopened" for consideration. In a like manner, most of the other elements may iteratively affect nearly any other element. It may happen that a particular SS OT&E result has such a large system impact that all "intermediate" levels are "skipped" to determine possible alternative mission

objectives. Of course the intermediate levels are not really skipped. Instead, the levels become merged over a short time period (this is popularly called "crisis management" or "putting out the fires"). This is done in order to make timely decisions to solve a problem which could have critical impact (cost, schedule, or technology) upon the complete system.

### 3.4 MEASURES OF SOFTWARE SUPPORTABILITY RISK.

There are several measures (or metrics) which are integral to software supportability risk assessment process.

- a) ECS SS Profile Metrics
- b) SS Evaluation Metrics
- c) SS Negative Outcome Probability Estimates
- d) SS Magnitude of Consequence Estimates
- e) SS Risk Levels
- f) Risk Agent Acceptance Levels

Figure 3-7 integrates these measures loosely with the risk assessment process to illustrate the risk measure derivation. The OT&E objectives and MOEs provide the force; the baseline support requirements and support evaluation metrics are the anchor in the process; the estimation metrics are derived from heuristics, experimental tests, or historical data using techniques in section IV; risk levels/metrics are simply integration of estimated negative outcome probability density functions over a specified range of outcomes; and the risk levels plus magnitude of consequence metrics are considered by the risk agent in determining the risk acceptance levels.

#### 3.4.1 ECS SS Profile Requirements Metrics.

The following categories create a maximum of 27 requirements:

- a) Priority Type:
  - E - Emergency
  - U - Urgent
  - N - Normal

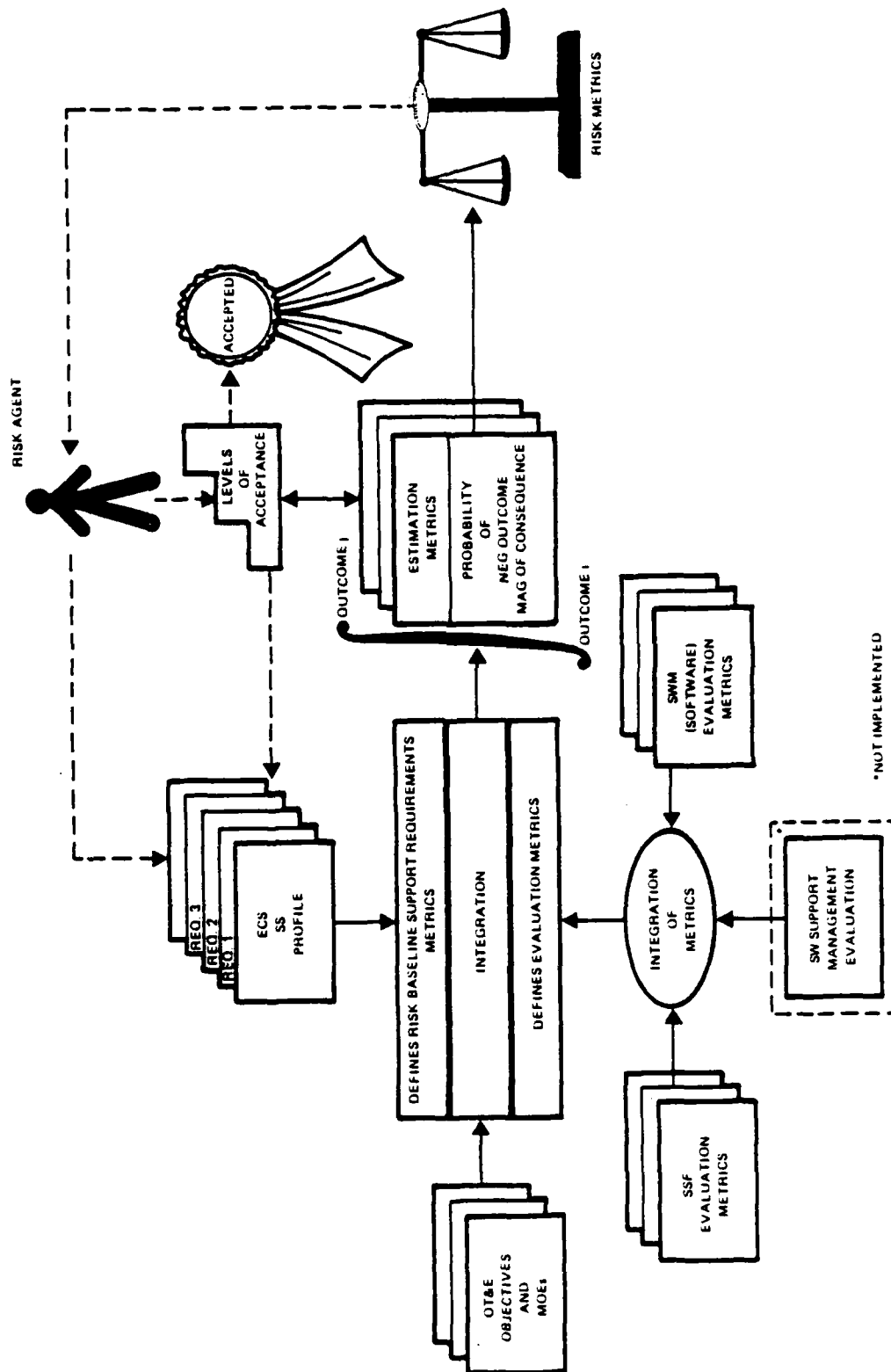


Figure 3-7. Software Supportability Risk Measure Derivation

- b) Maintenance Action Type:      C - Correction  
   A - Enhancement  
   V - Conversion
- c) Complexity Level:                L - Low  
   M - Medium  
   H - High

For example, the requirement for software support turnaround for an emergency low complexity corrective maintenance action (ELC) might be one hour. Other requirement metrics are not excluded from consideration in this profile.

#### 3.4 2 SS Evaluation Metrics.

The SS Evaluation Metrics include the current AFOTEC software support facility and software maintainability metrics as well as any new metrics developed in the future. A new set of software support management metrics for design and configuration management were suggested in section 3.3.2. The AFOTEC SS evaluation measures are described in reference 5.1.

#### 3.4.3 SS Negative Outcome Estimates.

Using techniques defined in section IV, an estimate of likelihood of occurrence for each negative outcome to be considered is made. Only the negative outcomes which might be expected to drive the OT&E software supportability risk acceptance should be considered. A family of probability density functions  $F_R = \{P_i : i=1..n\}$  (which may well be "standard" density functions as initial estimates) is derived for each class (k) of negative outcomes. Such a class might consist of all the  $P_i$  as the software maintainability evaluation metric is allowed to vary from 1 to 6 in discrete steps (with the set of fixed variables as in the example of section 3.3.1 3).

The use of classes and families of probability density functions is necessary only for sensitivity analysis for risk reduction and alternative choices and may only consist of two or three members. The family of probability density functions for the example in section 3.3.1.3 are of a Rayleigh curve form. Once the parametrization for the curve is determined, the specific family member shape is determined and the derivations of the estimation metrics is complete.

#### 3.4.4 SS Magnitude of Consequence Estimates.

These metrics are a function of the set of negative outcomes, the probability of negative outcome occurrence, and the risk agent's assessment. This latter assessment may be objective or subjective. Objective assessments are measured behavioral responses of the risk agent to the negative outcome consequence. Normally, this objective measurement is not possible for software supportability negative outcomes. One possible example of this would be measuring the reaction of a user to varying interactive response times because of software modifications. Subjective assessments by the risk agent based upon subjectively or objectively derived data (e.g., history of similar system) is more likely to occur.

The metrics may be represented on any of the possible scales such as (LO, MED, HI), (INCONVENIENT, DANGEROUS, CATASTROPHIC), and so forth.

#### 3.4.5 SS Risk Levels.

Integration (or summation if the probability density function is discrete) of the probability density function over a range of possible outcomes results in a specific software supportability risk level. In the example of section 3.3.1.3, if it is desired to determine the risk due to negative outcomes of greater than or equal to 3 hours and  $p$  is the probability density function, then the risk  $R$  is given by:

$$R = \int_3^{\infty} p dx$$



Assuming the density function is a Rayleigh curve (which it appears to resemble), we might obtain:

$$R = \int_3^{\infty} a e^{-bx^2} dx$$

where "a" and "b" are parameters which determine the curve shape.

This same technique can be used to determine risk level for the discrete case, even for use of scales such as (LO, MED, HI). The usual technique is to assign a numeric value on an even interval scale to each of the fuzzy linguistic terms and then use normal summation indexing.

#### 3.4.6 Risk Agent Acceptance Levels.

The acceptance level is simply that particular value of the risk level which is acceptable to a specific risk agent. Implicit in this acceptance is the acceptance of the probability of the negative outcome.

For the example of section 3.3.1.3, the probability of 0.4 that an emergency low complexity correction maintenance action will take one hour more than the required 24 may not be acceptable to the user risk agent. Likewise, by altering the curve shape parameter so that a user acceptable value of 0.3 is attained might be unacceptable to the supporter risk agent based upon the SSF and SWM evaluation measures. However, if the supporter can obtain one more person with systems analysis background, perhaps the SSF evaluation metric will be improved enough so that the latter probability value of 0.3 is also acceptable to the supporter risk agent.

Various techniques for reduction of risk or selection of alternative courses of action may be necessary before the iteration of all risk agents involved to acceptable levels of risk is attained.

### 3.5 REPORTING SOFTWARE SUPPORTABILITY RISK.

AFOTEC has a well-defined process for reporting results of an OT&E evaluation (reference 5.40). Figure 3-8 summarizes the possible types of

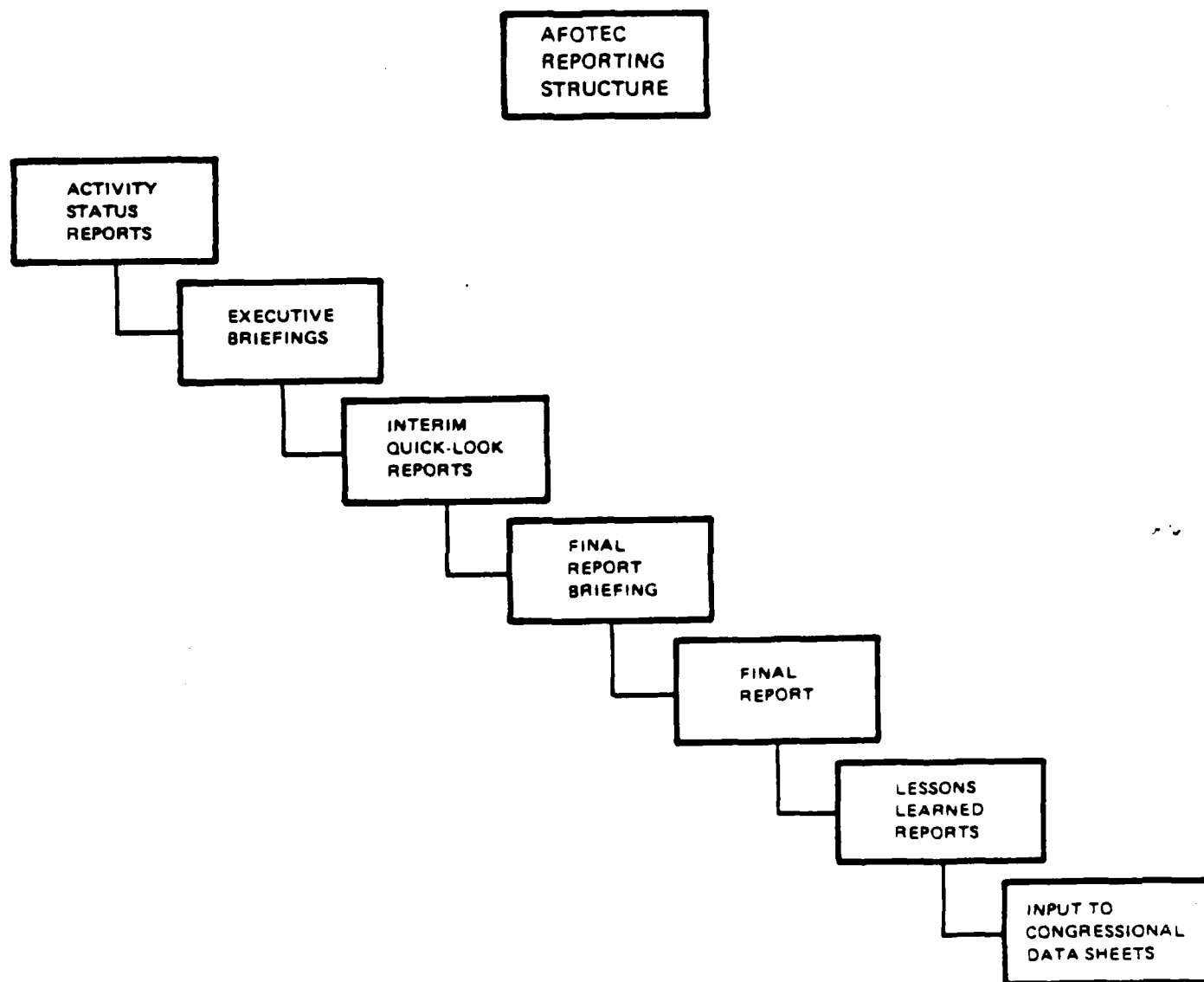


Figure 3-8. AFOTEC OT&E Reports

reports produced as part of OT&E. Since there are strict limits in preparation time, report length, and report content which are specified, any SS risk assessment results must be integrated into this more general OT&E report process.

The technical details covered throughout most of this report are inappropriate to include in most of the indicated reports except in excerpt form. Most of the technical details such as distribution functions, specific test factor component and component evaluation metrics, probability estimation techniques, and so forth, should be available as backup support. In case the summary presentation forms create concerns which require further explanation, the detailed information in computer or typed report form can be referenced.

The summary presentation form can take the form of graphs, tables, or matrices appropriately highlighted during briefings to show points of interest. The user/supporter risk matrix suggested in reference 5.12 is an excellent example of a risk report. Reference 5.1 and 5.41 contain tables which summarize the reporting of software maintenance evaluation metrics. Essentially, each of the metrics described in section 3.4 should be reported in a storyboard fashion. This technique would act as a summary of the risk assessment process, caveats and assumptions could be inserted as appropriate, and the key results such as evaluation metrics, risk levels, risk reduction and alternate choice analysis, and final acceptance levels with risk residues could be presented.

Such summary information could be easily included to varying levels of depth in each of the report types indicated in figure 3-8.

### 3.6 LEVEL OF EFFORT TO DEVELOP AND IMPLEMENT A CANDIDATE RAMSS.

Despite the fact that a methodology or technique has not been selected, it is possible to take a preliminary look at the level of effort required to develop and implement an RAMSS and the phasing of such an effort. Table 3-3 shows the anticipated phases, resources, and results for each phase.

Table 3-3.  
RAMSS Phased Development

<u>Phase</u>	<u>Function</u>	<u>Resources</u>	<u>Results</u>
I	Concept (Current Phase)	5 Months 3.5 People	RAMSS Framework RAMSS Potential Techniques RAMSS Integration with CSS OT&E Possible RAMSS Measures
II	Model Requirement Functional Spec. Document Pilot Study	6 Months 3 People	Requirements Specification RAMSS Design Model Risk Assessment Technique Output Form  RAMSS Procedures Manual: Data Collection, OT&E Evaluation, RA, and Reporting  Pilot Study Apply RAMSS to Current or Past OT&E Process  Assess Lessons Learned Procedures Automated Support
III	Upgrade Model Concept as Required From Pilot Study Develop and Implement Model Automated Sup- port Tools	8 Months 4 People	Automated Support Tools for RAMSS Selection of PDF Sensitivity Analysis Bookkeeping Interface to S/W Support Facility and S/W Maint Tools Automated Procedure Support

The reader will note it is proposed that the technique selected for development be experimented with manually before an automated model is developed and implemented. Due to the nature of the infancy of the

science of risk assessment of software, this approach should help to minimize the risk of developing a model which may not be completely feasible.

The estimates given above are not to be taken as final. The next report of this subtask will examine these estimates again as a function of the further analysis of candidate risk assessment measures.

**Section IV**  
**RA Methodologies, Techniques, Tools**

## SECTION IV

## RA METHODOLOGIES, TECHNIQUES, TOOLS TO SUPPORT AN RAMSS

## 4.1 TERMINOLOGY AND FOCUS.

The distinction among the terms methodology, technique and tool is not always clear. In fact, entities may encompass one or more of the terms. Generally a methodology is a broad-based collection of methods, rules, and postulates employed by a particular discipline. A technique is one of the methods of accomplishing a particular aim. A tool is something (manual or automated) which facilitates the application of a technique. Thus, a methodology is a discipline-based collection of techniques and tools. Since it is not possible to always make such a clear distinction as to which category a given entity belongs, this section briefly discusses specific entities which may be a methodology, technique, and/or tool for risk assessment. The term "methodologies" will be used henceforth for this general class of methodologies, techniques, and tools.

Since risk is the potential for realization of unwanted, negative consequences of an event, the risk assessment focuses upon a means to present that "potential." The primary means is as a probability. Determining the probability across possible negative consequences of an event, and across applicable events, results in a family of probability functions called probability density functions (PDFs). These PDFs may be discrete or continuous. There may also be some uncertainty in the actual PDF. The focus of risk assessment methodologies is upon determining a baseline PDF representative of the general risk function. Then, risk is defined when a measured or predicted negative outcome value is compared to the baseline density function. That is, risk is defined by those outcomes and their probabilities that are negative consequences with respect to the baseline. With this approach, ranges of risk such as "risky," "not risky," "high risk," "low risk," and so forth can be reasonably quantified.

Risk assessment methodologies generally rely on subjectively-derived data or objectively-derived data from the applicable area (e.g., software supportability). Subjectively-derived data depends on the cognitive ability of human beings to assign weights to possible outcomes of an uncertain event. Objectively-derived data depends on empirical observations/measurements and associated mathematical relationships among the measured (independent) and derived (dependent) variables which determines how weights should be assigned to outcomes of an uncertain event. No matter whether subjective or objective data is used, an underlying theoretical foundation for probability as applied to risk is necessary to interpret and determine sensitivity of the results.

This section provides a brief overview of the theoretical foundations of risk, and several of the subjective-based and objective-based risk methodologies. The information presented is an expansion of the information presented in section 4.3 of the reference 5.1 report.

#### 4.2 THEORETICAL FOUNDATION FOR RISK MEASUREMENT.

Risk is defined as "a possible negative outcome" (reference 5.30) or as "the realization of unwanted, negative consequences of an event" (reference 5.25). These definitions imply that the concept of risk is two-dimensional; i.e., risk consists of two parts. One part of risk is the negative outcome or the unwanted consequence. The second part of risk is the probability or potential of the negative outcome's occurrence. These two parts can be conveniently thought of and represented as two orthogonal scales as shown in figure 4-1(a).

Probability, the vertical scale in figure 4-1(a), is measured in conventional statistical terms. That is, the measure of probability ranges from 0 percent (no chance of occurrence) to 100 percent (absolute certainty of occurrence). A probability value is associated with each outcome. Outcome can be measured in a number of ways and depends on the problem context in which the risk assessment is being made. In the case of software supportability, outcome may be specified by either a cost,



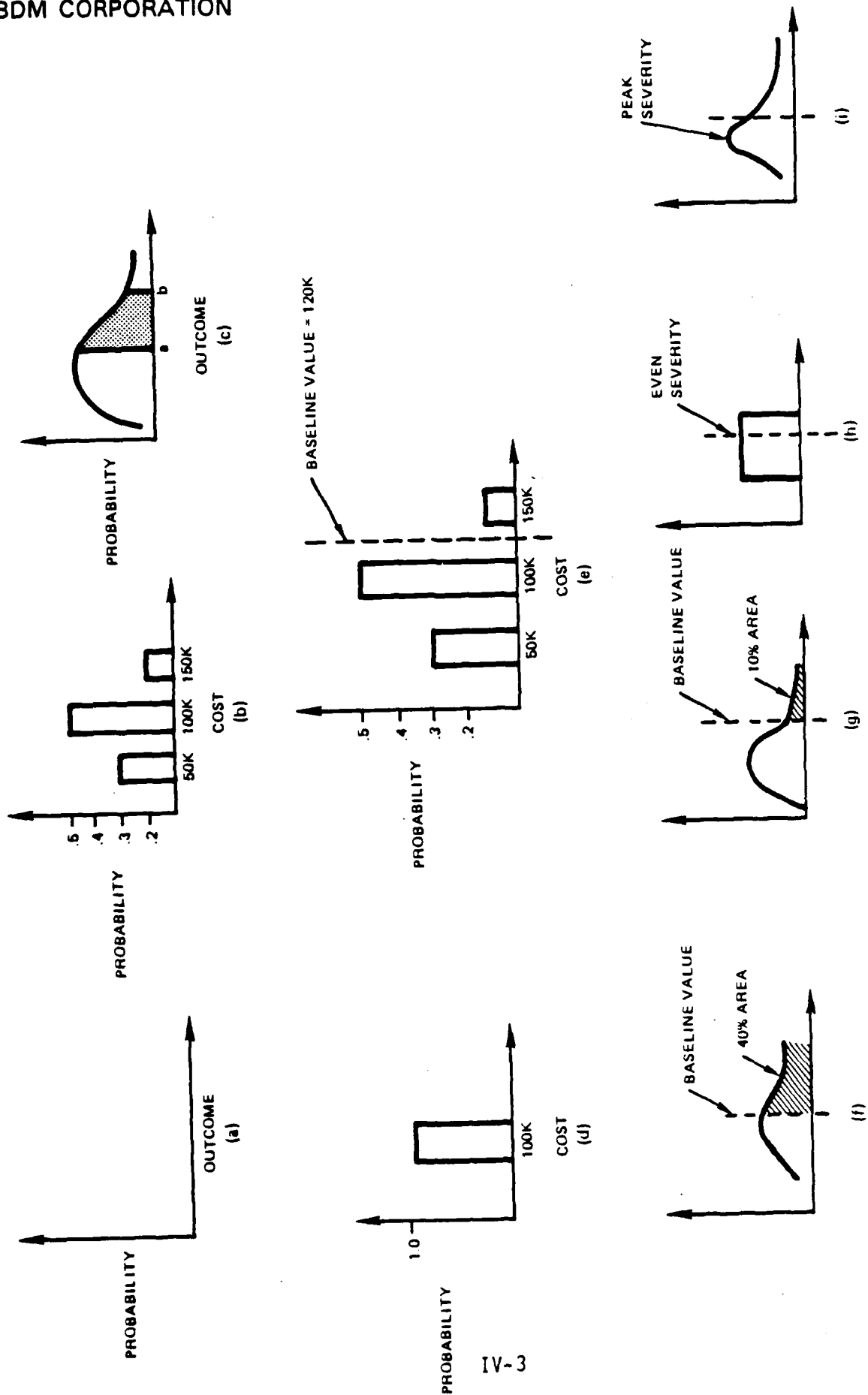


Figure 4-1. Theoretical Foundations of Risk

schedule, or performance variable. For example, consider that to support a given software package, it is estimated that there is a 30 percent chance that supportability will require 50,000 dollars, a 50 percent chance that 100,000 dollars will be needed for supportability, or a 20 percent likelihood that 150,000 dollars will be required. This case is depicted in figure 4-1(b).

When outcomes are assigned probabilities so that the probabilities add up to 100 percent, then a probability density function is established. The probability density function is a fundamental concept to risk assessment and its estimation is the basis for risk determination. Probability density functions may be discrete (as in the case of figure 4-1(b)) or continuous. For continuous probability density functions, the probability of occurrence for some interval of outcomes is that area under the density function that is cut off by the outcome interval. For example, in figure 4-1(c), the probability of an outcome greater than a and less than b is the area under the curve between a and b.

Implicit to the definition of risk is the notion of uncertainty. If there is no uncertainty, there is no risk. Risk analysts do not discuss situations with certain outcomes. Risk analysis specifically "attempt(s) to quantify uncertainty" (reference 5.26), and it is the probability density function that is the vehicle for the expression of uncertainty in quantitative terms. From the example depicted as figure 4-1(b), it is uncertain as to the cost of supporting a given software package. The uncertainty is expressed by explicitly stating that more than one cost outcome has a potential for occurrence. In other words, the cost of software supportability is not certain. Conversely, if it is certain that software supportability will require 100,000 dollars, as shown by figure 4-1(d), then there is no uncertainty in the risk.

Still to be considered in the definition of risk is the negative aspect, or magnitude, of the outcome. The concept of negative outcome or consequence can only be evaluated with respect to some baseline level. This baseline level is some value of the outcome which usually represents the available resources. For instance, if we are allocated 120,000

dollars for supportability and our estimation of outcomes are those shown in figure 4-1(b), then the negative outcome is that area of the probability density function that exceeds the baseline value. This relationship is shown in figure 4-1(e).

Given the conceptualization of risk put forth so far, then qualitative assessments of risk can be directly related to the area of the probability density function that exceeds the baseline value. Now terms such as "high" or "low" risk can be explicitly defined mathematically. As an example, high risk may be defined as a situation where 40 percent or more of the probability density function exceeds the baseline value (figure 4-1(f)). Low risk may be the case where 10 percent or less of the probability density function exceeds the baseline outcome (figure 4-1(g)).

Risk assessment must go further than simply considering the probability component of risk. The severity of the outcome has to be accounted for. To illustrate this idea, consider figures 4-1(h) and 4-1(i). In both, 30 percent of the probability density function exceeds the baseline value. However, it is apparent that figure 4-1(i) represents the riskier situation since the possible outcomes are more severe. Thus, risk is some combination of frequency (probability) and severity.

The key to risk assessment is the estimation of the probability density function. In other words, some estimate must be made of the outcomes (e.g., costs) and the probability of each outcomes' occurrence (perhaps dependent upon risk agent). It is this step in which the risk analyst must find a methodology which best conforms to the theoretical framework of risk just laid out. This step is usually an arduous task. Data sources for most risk assessments are quite limited. Thus, a risk assessment methodology is used that is practical, implementable, and can yield some evaluation of risk, however partial the analysis. Not every risk assessment methodology, however, explicitly or implicitly attempts to estimate a probability density function.

#### 4.3 RISK METHODOLOGIES, TECHNIQUES, TOOLS.

This section presents several risk assessment techniques (or methodologies) that have appeared in the literature. Each technique is described in a manner to give the reader a flavor of the technique's mechanics, advantages, and limitations. A comprehensive description of each methodology is not the purpose of this review. Complete details of each methodology can be found elsewhere: Atzinger and Brooks (reference 5.26); Megil (reference 5.27); Defense Systems Management College (reference 5.35); Apostolakis (reference 5.36); Behm and Vaupel (reference 5.37). Often entire books are devoted to a full description of some of the methodologies described in this section.

The following methodologies are described:

- a) Choice-between gambles technique
- b) Standard lottery
- c) Modified Churchman-Ackoff technique
- d) Delphi procedure
- e) Closed form questionnaires
- f) Bayesian analysis
- g) Network analysis
- h) Decision trees
- i) Parametric modeling
- j) Decision theory.

##### 4.3.1 Subjective Risk Techniques.

Risk assessment methodologies usually rely on either objectively-derived data or subjectively-derived data. First, let's consider methodologies using subjective data. Several methodologies exist in the literature for arriving at an estimated probability density function based on subjective judgments. These methods include: choice-between-gambles technique, lotteries, a modified Churchman-Ackoff technique, modified Delphi technique, Bayesian estimates, and estimates of the moments of the distribution via direct questioning. Several other risk

assessment methodologies exist that are based on subjective data. However, none of these other methods attempt to estimate a probability density function. These methods include checklists, qualitative surveys, rating scale surveys, and so on. In essence, these methods attempt to yield a "gut feel" of risk as opposed to an explicit statement of risk by a probability density function. Some of these techniques will be described in some detail as to the technique's mechanics, advantages, and limitations.

#### 4.3.1.1 The Difficulty of Making Subjective Estimates.

Making subjective estimates is a difficult task for humans to handle. In other words, subjective estimates place severe demands on the cognitive abilities of human beings. Because humans have limited abilities to process information and solve problems, estimation by subjective means must be evaluated in this light.

Because of the complexity and difficulty of many information-processing and decision-making situations, the human mind employs "heuristics." These heuristics or "rules of thumb" simplify the complexity of a given task. One heuristic that has been identified is anchoring and adjustment. An anchor is some original point estimate. For example, say that one is estimating the cost of a software package that is to be developed. One's expected cost may be 100,000 dollars. This initial anchor is now used to assess various other estimates of the cost, say the lowest possible cost or the highest possible cost. However, the original estimate of the expected value (100,000 dollars) is such a heavy anchor that adjustments around the point estimate are too small. That is, the lowest possible cost estimate and the highest possible cost estimate are too close to the "anchor" estimate. In other words, people's probability density functions are too tight. This holds even if the initial estimate is no more than a guess.

The estimation of a median value of some uncertain quantity displays other limitations to the human mind. People tend to focus their thinking on the unique situational features of the particular case facing them.

The information obtained in the actual outcomes of similar cases is often ignored. There is a tendency to mentally enhance some information of certain factors in order to increase the perceived uniqueness of the problem at hand. Each of us likes to believe that his/her own situation is unique. However, to pretend that one's own situation is so unique that it has no relationship to similar experiences is foolish.

Another heuristic employed by the human mind is availability. When an individual assesses the probability that an event will occur by imaging similar events or recalling related information, they are using the availability heuristic. The more "available" such related instances or information are, the higher the probability assigned to that event. As an example, if one saw a fatal car accident yesterday, then today their estimate of the probability of a fatal accident would be artificially high. The availability heuristic helps explain why the time or cost needed to complete a task is usually underestimated. There is, of course, an incentive for making a low estimate when doing the work for someone else, but even when people are making the estimate for themselves, the bias towards estimates that are too low still exists. The most "available" scenario is the "surprise-free" one--each subtask is easily completed in the minimum time--and thus the median estimate is usually derived from the no-complication scenario.

#### 4.3.1.2 Choice-Between-Gambles Technique.

The objective of the choice-between-gambles technique is to subjectively derive a discrete probability density function. The density function displays the probability that a component characteristic (e.g., a cost, schedule, or performance variable) will achieve a specified level. The inputs for eliciting these probability responses from a group of experts are composed of choice-between-gambles or betting-type questions administered by the analyst. It is believed by many authors in the field of subjective decision making, that this form of questioning results in a more realistic probability density function than a direct questioning

approach. This latter method of asking the expert directly what probability he/she attaches to a particular outcome, although simple in application, has little likelihood of success according to many authors. Many individuals either have no ability to think directly in terms of probability, or they have difficulty communicating the probabilities without the aid of some tool such as the choice-between-gambles technique.

#### 4.3.1.2.1 Description.

The choice-between-gambles technique is an iterative procedure which is initiated by presenting two alternative gambling situations to an expert. The expert is asked to choose between (1) a real-world gamble involving values of a component characteristic (e.g., cost) of the project in question with unspecified probabilities and (2) a hypothetical gamble involving two objective events, E1 and E2, with given probabilities,  $P(E1)$  and  $P(E2)$ . The monetary payoffs for both gambles are made equal to facilitate the expert's ability to discriminate. Next, the probabilities of the hypothetical gamble are varied (starting with equal probabilities for E1 and E2) until the expert is indifferent between the two gambles. Hence, the expert's subjective probabilities regarding the outcomes of the real-world gamble are inferred by the resulting probabilities from the hypothetical gamble.

As an illustration of this technique, consider a real-world gambling situation involving two possible costs for an avionics software package and a hypothetical gamble with possible events E1 and E2. The payoffs are stated as: (1) \$10 if one cost is realized, and \$0 if the other cost occurs; and (2) \$10 if event E1 occurs, and \$0 if event E2 is realized. Table 4-1(a) reflects this initial decision situation.

The assumption is that if the expert chooses the real-world gamble, he will receive \$10 if the software cost of 36,000 dollars (plus or minus 1,000 dollars) actually occurs. The expert will receive \$0 if any other cost is realized. If the expert selects the hypothetical gamble, he will

Table 4-1.

Example: Choice Between Gambles Technique

## (a) Decision Situation

Real-World Gamble			Hypothetical Gamble		
Payoff	\$10	0	Payoff	\$10	0
Cost	\$36,000	not \$36,000	Event	$E_1$	$E_2$
	\$+1,000	+1,000			
Probabilities	?	?	Probabilities	0.5	0.5

## (b) Revised Decision Situation

Real-World Gamble			Hypothetical Gamble		
Consequence	\$10	0	Consequence	\$10	0
Cost	\$36,000	not \$36,000	Event	$E_1$	$E_2$
	\$+1,000	\$+1,000			
Probabilities	?	?	Probabilities	0.7	0.3

## (c) Final Probability Distribution

Cost	Probability
\$32,000 + 1,000	0.0
\$34,000 + 1,000	0.2
\$36,000 + 1,000	0.7
\$38,000 + 1,000	0.2
\$40,000 + 1,000	0.0



receive \$10 if E1 occurs and \$0 if E2 occurs. Therefore, if the expert's decision in the first round is to accept the real-world gamble, then it is immediately inferred that his/her subjective probability assessment that a cost of 36,000 dollars (plus or minus 1,000 dollars) will be achieved is greater than 0.5. Thus, in the next decision rounds the analyst will adjust the probability of occurrence of the hypothetical event E1 upward, and that for event E2 downward. This procedure is then continued in an iterative fashion to the stage where the expert is indifferent to the two gambling situations. Suppose that this stage is ultimately achieved at  $P(E1) = 0.7$ ,  $P(E2) = 0.3$ . Then  $P(\text{software cost} = 36,000 \text{ dollars plus or minus } 1,000 \text{ dollars})$  is inferred to be 0.7. This revised decision situation is depicted in table 4-1(b).

Having obtained the probability of software cost equal to 36,000 dollars, the next step is to change the software cost in the real-world gamble to the next interval that the expert will be able to discriminate between its probability of occurring over that of the previous value. At each successive stage, then, probabilities are derived for various interval values of software cost. Finally, each endpoint of the probability density function is determined when the expert is indifferent between the two gambles, with  $P(E1) = 0$  and  $P(E2) = 1$ .

The resulting probability density function for this example could be shown as in table 4-1(c).

Notice in table 4-1(c) that the total of the subjective probabilities do not equal 1.0; instead the total is 1.1. Given this situation, then the analyst can either: (1) reassess the expert's subjective probabilities, or (2) normalize the derived probabilities so that the total equals 1.0.

#### 4.3.1.2.2 Advantages.

The choice-between-gambles technique derives probability density functions through inference rather than by direct questioning. As noted earlier in the discussion, it appears that such an organized stepwise

procedure for eliciting a probability density function results in a more valid output.

Compared to most other techniques, the choice-between-gambles technique is not time consuming in its application. It is simple to apply and results directly in a probability density function.

#### 4.3.1.2.3 Limitations.

First, the choice-between-gambles technique only results in discrete probability density functions. Continuous probability density functions cannot be obtained as the technique is described here. Second, the expert may find it difficult to determine the highest or lowest value for which he/she can state a subjective probability.

#### 4.3.1.2.4 Assumptions.

It is assumed that the monetary rewards offered as consequences for correct responses are sufficient in magnitude to motivate the expert in forming his/her judgements. It is also assumed that the expert's concern for the project success, his/her integrity, and his/her decision-making abilities contribute to the degree to which his/her judgements represent his/her personal beliefs. Finally, it is assumed that the so-called expert is in fact knowledgeable and experienced in his/her field. The expert must also be sufficiently well-founded in probability theory to respond meaningfully to the questioning procedure.

#### 4.3.1.3 The Standard Lottery.

The objective of the standard lottery technique is the derivation of a probability density function over all possible values of a given component characteristic (e.g., cost). The procedure involves presenting the expert with two gambling situations. This technique differs from the choice-between-gambles method in that it does not involve the process of

varying actual probabilities until indifference is achieved. Instead, numbers representing randomly selected lottery tickets from a batch of 100 are varied in an attempt to achieve indifference. In essence, the number of such tickets directly infers component risk probabilities.

#### 4.3.1.3.1 Description.

The standard lottery technique is based on the following basic lottery description. In a lottery, a contestant purchases as many tickets as desired. The more tickets he/she purchases, the greater his/her chance of winning the contest prize. After the purchase of tickets is completed, one number is randomly drawn from a lot of (for example) 100 equally likely numbers. That is, each contestant fully understands that any number between 1 and 100 has an equal chance of being selected. The winning contestant is that individual who owns a lottery ticket with the number on it coinciding with the number selected. For example, a contestant might have purchased 40 tickets; thus his/her chance of winning the contest prize is 0.4.

The standard lottery technique proceeds as follows:

- a) Specify a possible component characteristic value (e.g., the cost of an avionics software package is 100,000 dollars) for the real-world event.
- b) Direct the expert to imagining that he/she is given a choice between a certain number of tickets in the standard lottery with a prize of value  $V$  (e.g., \$10) and the right to receive the same prize if the real-world event is realized.
- c) For a given initial number of lottery tickets (e.g., 30) ask the expert which alternative gamble he/she feels has the greatest chance of winning the prize: (1) the realization of the real-world event, or (2) the holding of the specified number of tickets (i.e., 30) of a lottery of 100 tickets total.

- d) If one gamble is preferred over the other, then vary the given number of tickets and repeat step c.
- e) Repeat steps c and d until, in the expert's opinion, he/she feels that the possibility of receiving the prize in the event has exactly the same likelihood as some number of tickets (say 70) in the lottery. Thus, it is inferred that the expert considers that there is a 70 percent chance of the software cost being 100,000 dollars.
- f) Employing steps a through e, the expert can proceed analogously to assign probabilities to all other possible real-world events.

Similar to the choice-between-gambles technique, the probabilities must sum to 1.0. Normalization can be used in those cases where the probabilities do not exactly sum to unity.

#### 4.3.1.3.2 Assumptions, Limitations, Advantages.

The standard lottery technique also provides an improved process for eliciting subjective responses over direct questioning. The lottery technique is similar to the choice-between-gambles technique and thus offers the same advantages, limitations, and assumptions. Importantly, the expert with little probability theory background may be more comfortable with this technique as opposed to the choice-between-gambles method.

#### 4.3.1.4 The Modified Churchman-Ackoff Technique.

The modified Churchman-Ackoff technique differs from the previous two methods in that (1) there are no betting situations, and (2) the expert is instead asked to make "greater than," "equal to," or "less than" evaluations regarding relative probabilities. A resulting relative probability scale is easily transformed into a probability density function.

#### 4.3.1.4.1 Description.

In the modified Churchman-Ackoff technique, the expert must reveal a range of possible values which the component characteristic (e.g., the cost of a software package) could possibly realize. By employing some other technique, end point values of zero probability of occurrence must first be specified. These values need only be any low and high values which the expert specifies as having zero probability of occurrence in the proposed system.

Next, individual values within the range of possible values must be determined. These values, which will form the set of comparative values (e.g., cost values) for this technique, are specified by the following approach:

- a) Start with the smallest value.
- b) Progress upward on the scale of values until the expert is able to state a simple preference regarding the relative probabilities of occurrence of the two characteristic values. If he/she is able to say that he/she believes one value has either a greater chance or a lesser chance of occurring than the other of the two values, then it is inferred that the expert is able to discriminate between the two values.
- c) Using the higher of the two previously specified scale values as a new basis, repeat step b to determine the next value on the scale.
- d) Repeat steps b and c until the high end point value of the range of parameter values is approached.

Employing this procedure, one might obtain the results in table 4-2(a).

The descending order of probability of occurrence can be determined by applying the following paired comparison method.

Ask the expert to compare, one at a time, the first discrete value ( $\theta_1$ ) of the set to each of the other values ( $\theta_2$ ,  $\theta_3$ , etc.). The expert

Table 4-2.

## Example: Modified Churchman-Ackoff Technique

## (a) Characteristic Values for Software Cost

$$\theta_1 = \$35,000$$

$$\theta_2 = \$36,000$$

$$\theta_3 = \$37,500$$

$$\theta_4 = \$38,500$$

$$\theta_5 = \$40,000$$

$$\theta_6 = \$41,000$$

$$\theta_7 = \$41,500$$

## (b) Paired Comparisons

$\theta_3$ vs $\theta_4, \dots, \theta_7$	$\theta_4$ vs $\theta_5, \dots, \theta_7$	$\theta_5$ vs $\theta_6, \theta_7$	$\theta_6$ vs $\theta_7$
$\theta_3 < \theta_4$	$\theta_4 > \theta_5$	$\theta_5 > \theta_6$	$\theta_6 > \theta_7$
$\theta_3 > \theta_5$	$\theta_4 > \theta_6$	$\theta_5 > \theta_7$	
$\theta_3 > \theta_6$	$\theta_4 > \theta_7$		
$\theta_3 > \theta_7$			

## (c) Summary of Preference Relationships

$$\theta_4 = 6 \text{ times}$$

$$\theta_3 = 5 \text{ times}$$

$$\theta_5 = 4 \text{ times}$$

$$\theta_2 = 3 \text{ times}$$

$$\theta_6 = 2 \text{ times}$$

$$\theta_1 = 0 \text{ times}$$

$$\theta_7 = 0 \text{ times}$$

Table 4-2.

Example: Modified Churchman-Ackoff Technique (Continued)

## (d) Transformation

Characteristic Value	Preference Rank	New Symbol
\$38,500 $\Theta_4$	1	$X_1$
\$37,500 $\Theta_3$	2	$X_2$
\$40,000 $\Theta_5$	3	$X_3$
\$36,000 $\Theta_2$	4	$X_4$
\$41,000 $\Theta_6$	5	$X_5$
\$35,000 $\Theta_1$	6	$X_6$
\$41,500 $\Theta_7$	7	$X_7$

## (e) Relative Probability Ratings

$RX_1$  = 100 probability points  
 $RX_2$  = 80 probability points  
 $RX_3$  = 50 probability points  
 $RX_4$  = 25 probability points  
 $RX_5$  = 10 probability points  
 $RX_6$  = 0 probability points  
 $RX_7$  = 0 probability points

Table 4-2.

Example: Modified Churchman-Ackoff Technique (Concluded)

## (f) Probability Density

Component Characteristic Value	Probability
$x_1$	0.377
$x_2$	0.301
$x_3$	0.189
$x_4$	0.095
$x_5$	0.038
$x_6$	0.000
$x_7$	<u>0.000</u>
Total	1.000



is asked to state a preference for that value in each group of two values that he/she believes has the greater chance of occurring. In other words, the expert chooses one value which has the greatest chance of occurrence for each paired comparison. The following hypothetical preference relationships could result for the set of 7 values:

$$\{\theta_1 < \theta_2, \theta_1 < \theta_3, \theta_1 < \theta_4, \theta_1 < \theta_5, \theta_1 < \theta_6, \theta_1 < \theta_7\}.$$

Next, ask the expert to compare, one at a time, the second discrete value ( $\theta_2$ ) of the set to each of the other values succeeding it in the set (i.e.,  $\theta_3, \theta_4$ , etc.). The following preference relationships might result:

$$\{\theta_2 < \theta_3, \theta_2 < \theta_4, \theta_2 < \theta_5, \theta_2 < \theta_6, \theta_2 < \theta_7\}.$$

Continue the process until all values ( $\theta_j$ ) have been compared to the others. For example, table 4-2(b) lists preferences which might result for the remaining cost values.

Now total the number of times ( $\theta_j$ ) value was preferred over other values. The results for this procedure are listed in table 4-2(c).

List the values in descending order of simple ordinal probability preference and change the symbols for each value from  $\theta$  to  $X(j)$  as shown in table 4-2(d).

Arbitrarily assign a rating of 100 points to the characteristic value (e.g., cost) with the highest subjective probability. Then, as in the first step, question the expert regarding the relative chance of occurrence of each of the other values on the ordinal scale in table 4-2(d) with respect to the value at the top of the scale. Assigning  $X(1)$  a rating of 100 points, the expert is first questioned as to his/her feeling of the relative chance of occurrence of the second highest scale value (e.g.,  $X(2)$ ) with respect to  $X(1)$ ). Does it have a 25 percent as much chance of realization as  $X(1)$ ? A 60 percent? A 70 percent? The

relative probability rating, based on 100 points, (i.e., 100 percent as much chance) will then be posted for  $X(2)$ . For example, if the expert decides that  $X(2)$  has 8/10 as much chance of occurring as does  $X(1)$ , the ratings become  $X(1) = 100$  points and  $X(2) = 80$  points.

Next, question the expert about the relative chance of occurrence of the next highest scale (e.g.,  $X(3)$ ), first with respect to the most preferred value ( $X(1)$ ), and second with respect to the second most preferred scale value ( $X(2)$ ). The resulting numerical ratings should concur.

If the expert expresses a belief that  $X(3)$  has 1/2 as much chance as  $X(1)$  and 5/8 as much chance as  $X(2)$  (as a validity check), this confirms that the relative probability of occurrence rating for  $X(3)$  is 50. The scale now is  $X(1) = 100$  points,  $X(2) = 80$  points, and  $X(3) = 50$  points.

The process is continued for each remaining successively lower scale value on the ordinal scale shown in table 4-2(d). Determine the relative number of points to be accorded each value with respect to the top scale value and with respect to all other values on down the scale which are above the characteristic value in question.

In the event of minor disparities between relative probability ratings for a given value, the average of all such ratings for that characteristic value (e.g., cost value) might be computed. For example,  $X(4)$  might be determined to be 3/10 as probable as  $X(1)$ , 1/4 as probable as  $X(2)$ , and 1/2 as probable as  $X(3)$ . The three absolute ratings for  $X(4)$  are thus inferred to be 30, 20, and 25 points respectively. The average of these ratings is 25. However, before averaging such figures, it might be beneficial to have the expert reevaluate his relative ratings for  $X(4)$  with respect to  $X(1)$ ,  $X(2)$ , and  $X(3)$ .

As a result of the above process, the relative probability values shown in table 4-2(e) might be attained.

Finally, the scale of relative probability values can be converted directly into a scale of actual probability density values by letting  $P(X_1)$  equal the actual subjective probability of occurrence of the highest value. Then,  $P(X_2)$  is then defined as

$$\frac{R(X_2)}{R(X_1)} P(X_1).$$

Similarly  $P(X_i)$  is defined as

$$\frac{R(X_i)}{R(X_1)} P(X_1).$$

for  $i = 2, 3, \dots, 7$ .

Assuming that the independent characteristic values evaluated represent all possible values attainable by the component characteristic, the respective probabilities must sum to 1.0 (i.e.,  $P(X_1) + P(X_2) + P(X_3) + P(X_4) + P(X_5) + P(X_6) + P(X_7) = 1.0$ ). Substituting the expressions for  $P(X_i)$ ,  $i = 2, \dots, 7$ , it follows that

$$\begin{aligned} P(X_1) + \frac{R(X_2)}{R(X_1)} P(X_1) + \frac{R(X_3)}{R(X_1)} P(X_1) + \frac{R(X_4)}{R(X_1)} P(X_1) \\ + \frac{R(X_5)}{R(X_1)} P(X_1) + \frac{R(X_6)}{R(X_1)} P(X_1) + \frac{R(X_7)}{R(X_1)} P(X_1) = 1. \end{aligned}$$

Solving this equation for  $P(X_1)$ , the remaining  $P(X_i)$ ,  $i = 2, \dots, 7$  can be determined using the relationship

$$P(X_i) = \frac{R(X_i)}{R(X_1)} P(X_1).$$

As an illustration, consider the relative probability ratings in table 4-2(e). Using these values, the preceding equation is given by

$$P(X_1) + \frac{30}{100} P(X_1) + \frac{50}{100} P(X_1) + \frac{25}{100} P(X_1) + \frac{10}{100} P(X_1) = 1.$$

Solving this equation,  $P(X_1) = 0.377$ .

This value can be used to determine the remaining probabilities as follows:

$$P(X_2) = \frac{RX_2}{RX_1} P(X_1) = 0.80(0.377) = 0.301$$

$$P(X_3) = \frac{RX_3}{RX_1} P(X_1) = 0.50(.0377) = 0.189$$

$$P(X_4) = \frac{RX_4}{RX_1} P(X_1) = 0.25(0.377) = 0.0095$$

$$P(X_5) = \frac{RX_5}{RX_1} P(X_1) = 0.10(0.377) = 0.038$$

$$P(X_6) = \frac{RX_6}{RX_1} P(X_1) = 0(0.377) = 0.000$$

$$P(X_7) = \frac{RX_7}{RX_1} P(X_1) = 0(0.377) = 0.000$$

The resulting probability density appears in table 4-2(f).

#### 4.3.1.4.2 Advantages.

The modified Churchman-Ackoff technique offers an alternative to the two previous methods of eliciting absolute subjective probability responses. In this case, relative probabilities with respect to one chosen most probable characteristic value are derived. In some situations, the expert may think it easier to make evaluations with respect to a characteristic state that he/she feels has the greatest possibility of realization.

In addition, this technique offers a systematic method of checking the consistency of relative value judgements made by the experts. This enhances the validity of the resulting probability distribution function.

#### 4.3.1.4.3 Limitations.

The modified Churchman-Ackoff technique does not involve betting situations which are generally considered more successful in eliciting

correct responses. Instead, it involves an untested approach of directly eliciting relative percentage chances of occurrence statements for each value with respect to the occurrence of other characteristic values (e.g., does a software cost of 100,000 dollars have half as much, or 70 percent, or 90 percent as much chance of occurring as 120,000 dollars).

As with the other techniques discussed so far, the probability values are still judgements. That is, of course, the limitation of all techniques involving subjective (as opposed to objective) decision making.

#### 4.3.1.5 The Delphi Procedure.

Historically, the approach for obtaining a group consensus has been the formation of committees, commissions, or councils. While the basic philosophy may be sound, committees tend to pressure individuals into conforming. In addition, all opinions may not be expressed because of the personalities of the individuals and/or because of the relationship of the individuals within the group. Another drawback of committees is the tendency to spend a great deal of time discussing irrelevant issues. Further, irrelevant information may degrade the group's opinion. More serious though, is the possibility of a complete breakdown of the committee. That is, there is an inability of the committee to arrive at a general consensus.

The Delphi procedure is an alternative to the committee approach for eliciting a group judgement. The Delphi method attempts to improve the panel or committee approach in arriving at a forecast or estimate by subjecting the views of individual experts to each other's criticism in ways that avoid face-to-face confrontation. Thus, there is anonymity of opinions and of arguments advanced in defense of these opinions. Direct debate is replaced by the interchange of information and opinion through a carefully designed sequence of questionnaires. The participants are asked not only to give their opinions, but the reason for the opinions. At each successive questioning session, the experts are given new and

refined information, in the form of opinion feedback, which is derived by a computed consensus from earlier parts of the program. The process continues until further progress toward a consensus appears to be negligible. The conflicting views are then documented along with the consensus.

In sum, the primary features of Delphi procedures are:

- a) Anonymity of the sources of information among experts.
- b) Iteration with controlled feedback of group responses from iteration to iteration.
- c) Statistical group response (e.g., a median value).

#### 4.3.1.5.1 Description.

The steps of the procedure for estimating a group probability density function are outlined below for the cost of an avionics software package.

Employing the first two steps of the modified Churchman-Ackoff technique, each expert is asked to reveal his estimate of the total range of values which the software cost could realize. The individual values within this range will form the sets of comparative cost values. Then list all cost values specified by all of the experts. These will form the list of cost values to be investigated, for example, as those shown in table 4-3(a).

The list of characteristic values (e.g., cost values) to be investigated are included in table 4-3(b).

In the first round, randomly select a cost value from the list in table 4-3(b) and ask each expert to give an independent estimate of its probability. Each expert is questioned alone. In addition, each expert is asked his/her reasons regarding the probability assessment.

Arrange the probability responses from all experts in order of magnitude, and determine its quartiles, Q1, M, Q3, so that approximately one quarter of all estimates lie in each interval. For example, for the selected software cost of 31,000 dollars, the probabilities for experts E1, E2, E3, E4, and E5 might occur as shown in table 4-3(c).

Table 4-3.

Example: Delphi Procedure

## (a) Possible Characteristic Values by Experts

Expert 1 (\$)	Expert 2 (\$)	Expert 3 (\$)	Expert 4 (\$)	Expert 5 (\$)
28,000	29,000	30,000	29,000	30,000
31,000	31,000	32,000	30,000	32,000
32,000	33,000	34,000	32,000	34,000
34,000	36,000	37,000	33,000	36,000
37,000	40,000	39,000	36,000	37,000

## (b) List of Possible Discrete Characteristic Values

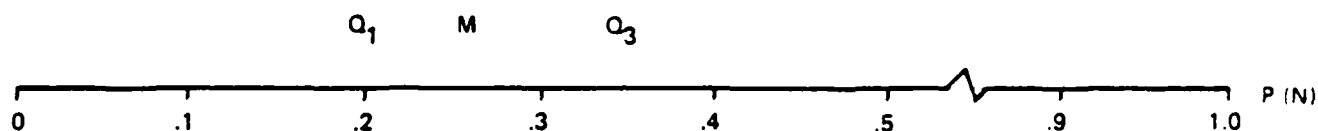
$Y_1 = \$28,000$   
 $Y_2 = \$29,000$   
 $Y_3 = \$30,000$   
 $Y_4 = \$31,000$   
 $Y_5 = \$32,000$   
 $Y_6 = \$33,000$   
 $Y_7 = \$34,000$   
 $Y_8 = \$36,000$   
 $Y_9 = \$37,000$   
 $Y_{10} = \$39,000$   
 $Y_{11} = \$40,000$

# THE BDM CORPORATION

Table 4-3.

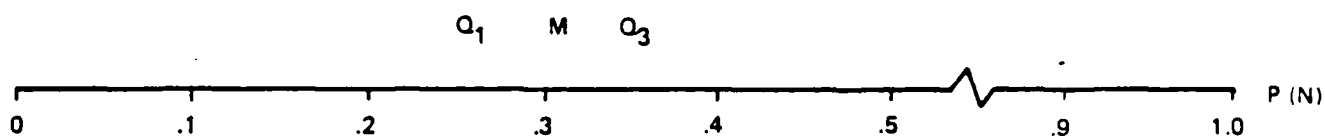
Example: Delphi Procedure (Concluded)

EXPERT	E <sub>4</sub>	E <sub>2</sub>	E <sub>1</sub>	E <sub>5</sub>	E <sub>3</sub>
PROBABILITY	.15	.25	.3	.325	.4



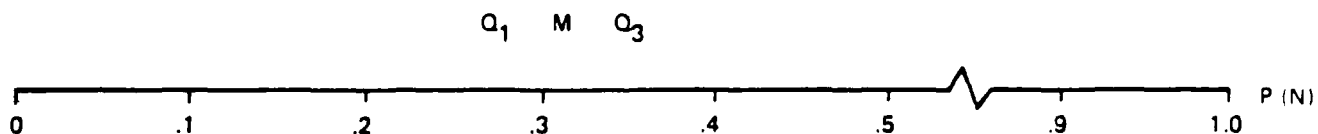
(c) PROBABILITY RESPONSES: 1ST ROUND

EXPERT	E <sub>4</sub>	E <sub>2</sub>	E <sub>1</sub>	E <sub>5</sub>	E <sub>3</sub>
PROBABILITY	.25	.275	.3	.325	.35



(d) PROBABILITY RESPONSES: 2ND ROUND

EXPERT	E <sub>2</sub>	E <sub>4</sub>	E <sub>1</sub>	E <sub>5</sub>	E <sub>3</sub>
PROBABILITY	.275	.275	.3	.325	.325



(e) PROBABILITY RESPONSE: 3RD ROUND



Reveal the values and responses of each interval to each member, and if his/her estimate lies outside the first round interquartile range, Q1 to Q3, then ask the expert to state reasons why the answer should be lower (or higher) than that of the 75 percent majority opinion expressed in the first round.

Give these new responses back to all respondents by communicating the new range of the new quartile values, along with independently stated reasons for the estimates outside the 75 percent majority opinion. The experts are now asked to consider the reasons given, weigh their feasibility, and revise their own previous estimates accordingly. For the software cost example, assume that the second round scale is as shown in table 4-3(d).

If the newly revised probabilities still fall outside the second round interquartile range, respondents are asked to state why they found previous arguments unconvincing enough to draw them toward the median.

In the third round, the quartile results of round 2 are submitted to respondents along with the counter arguments elicited. These respondents are then asked to make a final revision of their estimates.

The mean value of the resulting round 3 estimates (table 4-3(e)) is taken as the group response as to what the subjective probability consensus for the software cost value should be. For this example, the mean third round subjective estimate for a cost of 31,000 dollars is

$$2(0.275) + (0.3000) + 2(0.325) / 5 = 0.300 = P(\text{Cost} = \$31,000)$$

Now repeat the procedure for a second possible cost value. Normalize the distribution if necessary.

#### 4.3.1.5.2 Advantages.

In situations where one wants to use group judgement to analyze uncertainty, the Delphi procedure provides an alternative to the committee approach in the identification and consolidation activities of a

risk analysis. The Delphi method attempts to improve upon the committee approach by allowing the exchange of information in an environment that reduces the group pressure to conform. Also there is the removal of the impact of the dominant individual.

#### 4.3.1.5.3 Limitations.

Perhaps the biggest drawback in applying the Delphi procedure at this time is that very few analysts have experience in using the technique. In particular, there is a lack of training in preparing questionnaires and analyzing the results. One should not discount the importance of such training. If the questionnaire is prepared by unqualified people, the answers to the questions may be biased or the questions themselves may not really address the problem. In addition, since the procedure has had limited exposure, it may not be accepted immediately.

Another important consideration in the selection of the Delphi procedure is time; both time available for conducting the analysis and time involved in applying the procedure. Clearly, if there is little time available for developing a consensus of opinion, then the Delphi technique may not be a viable alternative. Next, one should consider whether the group responses can be aggregated meaningfully. If there is doubt about combining the group responses, then one would probably not want to use the Delphi procedure. Finally, one must consider whether there are any popular opinions to which there may be pressure to conform. This may influence the committee chairman to pressure the group or lead the group in the direction of a favored policy or opinion, even though it may not represent the group's opinion.

#### 4.3.1.6 Closed Form Questionnaire Technique.

The use of questionnaires completed by knowledgeable evaluators is a technique for collecting information covering a wide range of possible functions. For our purposes, the information might be for round 1 of a

Delphi procedure for assigning risk measures of effectiveness or risk probability density functions. The information may be part of a pre-evaluation targeting for software areas needing more test and evaluation scrutiny. Closed form questionnaires is a technique used by AFOTEC to obtain evaluation measures of software supportability (reference 5.1).

#### 4.3.1.6.1 Description.

The characterizing elements of a questionnaire are the response scale, evaluator sample/characteristics, and the question organizational structure. Validity of a questionnaire depends upon many variables, but the ability to repeat the questionnaire used under identical circumstances and obtain the same results is important. In order for conclusions to be reached, evaluators must also "reasonably" agree on the values to be assigned to individual questions.

Questions within only a precise selection of responses are termed closed form questions. For example, multiple choice questions, true-false questions, and in general questions requiring a response within a lower and upper bound on a linear scale are closed form questions. Essay questions or questions allowing for explanation are open form questions. AFOTEC has used a response scale as indicated below for its questionnaire statements where "agreement" is good and "disagreement" is bad. Table 4-4 is an example of one of AFOTEC's question statements and guidelines on how to interpret the scale.

- a) Completely agree
- b) Strongly agree
- c) Generally agree
- d) Generally disagree
- e) Strongly disagree
- f) Completely disagree.

Normally questions are answered by one or more persons called evaluators. The capability of each evaluator to respond accurately depends upon the knowledge of the evaluator in the subject area and upon the

Table 4-4.

## Example: Closed Form Question

Question Number S 62

QUESTION: The number of expressions used to control branching in this module is manageable.

CHARACTERISTIC: Simplicity (size simplicity).

Explanations: The count of control expressions is closely related to the number of independent cycles in a module. The more control expressions there are the more complex the control logic tends to be.

EXAMPLES: The following examples indicate how to count the control expressions:

<u>CONTROL STRUCTURE</u>	<u>STATEMENT</u>	<u>CONTROL EXPRESSION</u>	<u>COUNT</u>
<u>Decision</u>	IF (A. OR. B) GO TO 10	A:B	2
	IF (A. AND. B) GO TO 10	A:B	2
	IF (C.GT.D) GO TO 10	C.GT.D	1
	IF (A. AND.B). OR. (C.GT.D))		
	GO TO 10	A:B:C.GT.D	3
	CASE (I) OF	I=1:I=2:I=3	2
	1: A	(Alternatives)	(number of alternatives less one)
	2: B		
	3: C		
	END CASE		
<u>Iteration</u>	DO 10 I=1, 10	I.LT.1	
	A	I.GT.10	2
	10 CONTINUE		

GLOSSARY: Control Expression: IF, CASE, or other decision control expression. DO, DO-while, or other iterative control expression.

SPECIAL RESPONSE INSTRUCTIONS: The following guidelines will anchor A and F responses, but are fairly subjective (especially the F anchor). The guidelines for the A response is (sic) suggested from other independent research. Remember to count all repetitions of the same control expression also.

Answer A if count  $\leq 10$

Answer F if count  $\geq 50$

clarity with which the question is stated. If more than one evaluator is used, there is a risk that there may not be a consensus in their responses. If only one evaluator is used, natural evaluation bias may not give an accurate result. Frequently a question is really more than one question or the terms in the question may be misleading or undefined. This can lead to evaluation error due to lack of question reliability.

Questionnaires can be structured so that groups and subgroups within groups address particular functional parts of the general subject area. In this case, the way in which question responses are aggregated is of importance as well as the consistency of response scales across subgroups and groups. Also, it is a concern whether a group may have too much or too little "natural" weight due to the number of questions within the group. Weights are frequently assigned by users to groups, subgroups, or individual questions as a subjective level of importance or as a derived regression coefficient for use in the aggregation of hierarchy of evaluation values. Sometimes the structure of the questionnaire is similar to a decision tree where certain paths are taken on the basis of responses to particular branching mode questions. This allows for a generic discrimination of what parts of the subject are useful to cover by the questionnaire as well as exploring particular areas in more depth.

#### 4.3.1.6.2 Advantages.

Closed form questionnaires can serve as valuable checklists and guidelines over a broad range of a subject matter. Properly structured they can quickly pinpoint subject areas which are very poor or very good, and those areas needing more detailed analysis. These type of questionnaires are quite flexible and can be easily tailored to particular special cases. The questionnaires can form the initial data point for several of the other subjective (and even parametric objective) risk techniques including the modified Churchman-Ackoff technique and the Delphi procedure.

#### 4.3.1.6.3 Limitations.

The limitations of closed form questionnaires depends upon the desired use of the responses. The most frequent limitation is lack of detail in the response and the subjective nature of the response. The scale of measure chosen has a great influence upon the response. Particular linguistic words (even those dry ones such as completely agree) can evoke evaluator bias through unfavorable or favorable connotations. Most often, equal interval scales are selected and desired, but the responses do not fit an equal interval scale.

Questions may be very difficult to answer if proper support tools are not available. For example, determining the extent of module calling relationships without an automatically produced module call cross-reference list is very time-consuming for a large software system.

The disparity among evaluator responses may not allow for meaningful conclusions, especially for a few number of evaluators. The sample of evaluators used may not be representative of the general population of evaluators. Thus, the evaluation may not be repeatable. Evaluators normally have bias. It usually requires automated capabilities to process evaluator responses, determine authors, allow for selective elimination of evaluator responses, and aggregate questionnaire response values. Without automated support, the flexibility of using questionnaires across a range of applications is limited.

Since questionnaires are completed by evaluators in a manual manner, it can take a long time to complete a questionnaire. The utility of a more detailed questionnaire needs to be carefully assessed against the derived benefit. Frequently, it is not possible to vary the depth of the questionnaire and still obtain representative meaningful results.

Most of the limitations derive from lack of proper questionnaire design, evaluator expertise, and/or lack of proper procedures to complete and process the questionnaire responses.

#### 4.3.1.7 Bayesian Analysis.

A Bayesian believes that it is possible at any time to express one's state of knowledge about some variable (e.g., cost) in the form of a probability density function. As additional experimental evidence becomes available, Bayes' Theorem is used to combine this evidence with previous probability density functions in order to obtain a new posterior probability density function. For example, new cost estimates recently obtained may be added to a historical data base of costs. The new probability density function represents the updated state of knowledge.

##### 4.3.1.7.1 Description.

Consider the Bayesian analysis of  $p$ , an unknown parameter of a postulated probabilistic model of a system. Assume that the experimental outcomes with the system can be treated as the values of a random variable  $X$ , the characteristic of interest (e.g., cost). Based on past experience and all other available information, the Bayesian approach begins with the specification of a prior probability density function  $f_p(p)$ . The prior probability density function (PDF) reflects the analyst's prior beliefs about the value of the parameter  $p$ . The assumed model specifies the probability density function for the sample value of the characteristic  $X$ , given the value of the parameter  $p$ . Since  $p$  is being regarded as another random variable, the PDF for the sample value of  $x$  with parameter  $p$  is written as the conditional PDF.

$$f_{x|p}(x_0|p_0) = \text{Conditional PDF for the sample value of characteristic } x, \text{ given that the value of parameter } p \text{ is equal to } p_0.$$

Each time an experimental value of characteristic  $x$  is obtained, the continuous form of Bayes' Theorem, listed here,

$$f_{p|x}(p_0|x_0) = \frac{f_{x,p}(x_0,p_0)}{f_x(x_0)} = \frac{f_{x|p}(x_0|p_0) f_p(p_0)}{\int_{p_0} f_{x|p}(x_0|p_0) f_p(p_0) dp_0}$$

is used to obtain a posterior probability density function  $f_{p|x}(p_0|x_0)$  representing the analyst's new state of knowledge about the value of the parameter  $p$ . This posterior probability density function serves as the basis for any present decisions and also as the prior distribution for any future experimentation.

#### 4.3.1.7.2 Advantages.

In risk analysis, situations frequently exist where the analyst has available both objective test data and other relevant information based on the externalities of the problem. Often, due to cost and time constraints, there is only a limited amount of relevant test data available by the decision date. Thus, other factors such as previous test data, engineering judgment, experience with similar systems, etc., must be taken into consideration. In the context, Bayesian statistics provides the analyst with a tool for synthesizing this information into one probability distribution which can then be used directly to estimate the risks in question.

#### 4.3.1.7.3 Limitations.

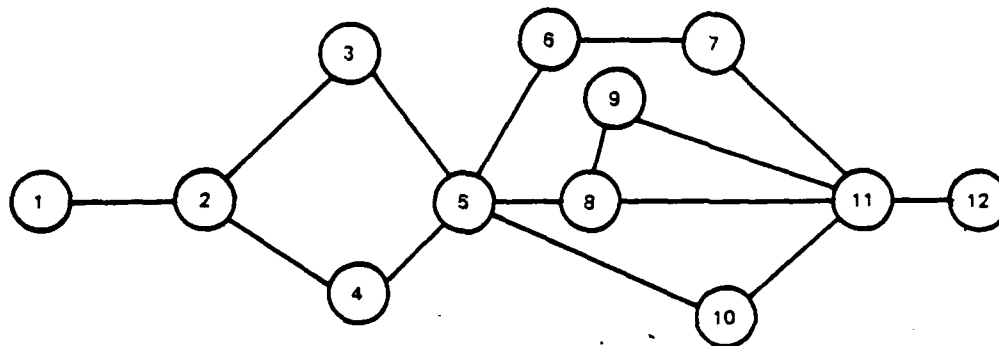
Unfortunately, there seems to be some mystique that surrounds any application of Bayesian statistics. This is due in some instances to a disagreement with the Bayesian philosophy and in others to the lack of true understanding of the mechanism of the Bayesian approach. Further, the mathematics of Bayesian analysis are also fairly complex. Advanced graduate training in mathematics or statistics is usually necessary to implement this technique. Perhaps one of the most widely used arguments against the use of the Bayesian procedure is the apparent absence of a rational basis for constructing a prior distribution.



#### 4.3.1.8 Network Analysis.

##### 4.3.1.8.1 Description.

Most people today are familiar with the concept of a network and network modeling. The figure below is an example of a network.



NETWORK

In such a network, each circle represents a decision point, event, or milestone, and each line represents an activity that must be finished to advance the program, that consumes resources, or that takes time. Network analyses such as PERT (Program Evaluation and Review Technique) have been used to manage schedule risk by establishing the shortest development schedules through the network, by monitoring and projecting program progress, and by funding and applying necessary resources for maintaining the schedule. Successors to PERT have estimated the minimum cost path through the network.

Numerous network models and network programming languages have developed in recent years. Several of these are quite sophisticated (see Pritzker and Pegden, reference 5.38). In current network modeling, the

network is defined, and for each activity, cost or schedule information is described in a probabilistic manner. Then, by using computers to simulate a large number of program completions through the network, the characteristics of the network can be examined.

During simulations, the events occur according to the probabilities that they were assigned. The probabilities are described using a mean value and variance to describe, say, the time of completion for each event in the network. The estimates of the mean and variance are based on subjective estimates of experts. More specifically, probability density functions are arrived at for each event by:

- a) Assuming some distributional form for the PDF for each event.
- b) Asking individual experts to determine for a given parameter, say cost, an optimistic value, denoted "a", a pessimistic value, denoted "b", and a most likely value, denoted "ml" for the values for each event in the network.
- c) The expert judgements are then combined into one estimate of the mean and variance of the value for that particular event.

$$\text{Mean} = (a + 4ml + b) / 6$$

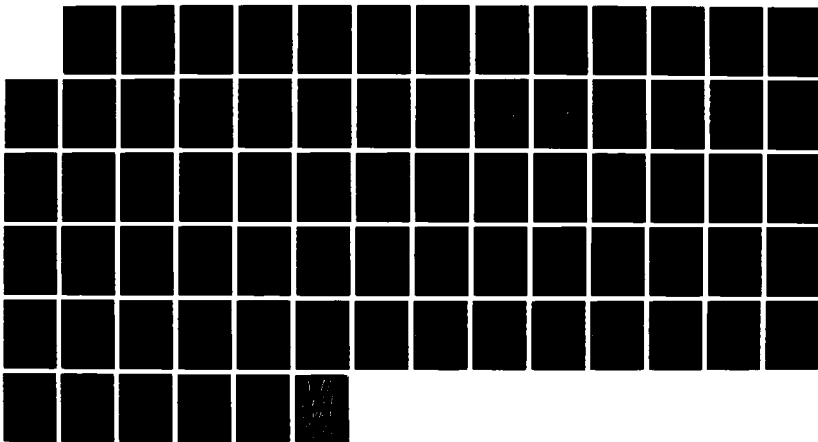
$$\text{Variance} = ((b - a)/6) ** 2$$

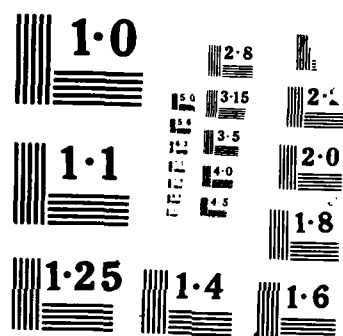
Network model outputs typically include some summary of the thousands of simulations that are possible using modern computing techniques. These summaries can be expressed as probability density functions or as statistical measures of a particular value in question, such as cost. Also, many of the network modeling languages can be used to describe minimum paths through the network. For instance, a least cost path, on average value, could be defined.

#### 4.3.1.8.2 Advantages.

The obvious advantage is that networks can now be evaluated quite thoroughly using the power of modern computers and recent advances in the

AD-A191 073 SOFTWARE SUPPORTABILITY RISK ASSESSMENT IN OTAE 2/2  
(OPERATIONAL TEST AND EVA. (U) BDM CORP ALBUQUERQUE NM  
W HUEBNER ET AL. 31 AUG 84 BDM/A-84-496-TR  
UNCLASSIFIED F29601-80-C-0035 F/G 12/5 NL





design of simulation languages. The thousands of possible combinations of event occurrences are revealed using this technique. In other words, the tendency of humans to imagine scenarios in which only a few factors vary is circumvented. A myriad of interactions within the system are examined.

#### 4.3.1.8.3 Limitations.

First and most importantly, the particular problem being analyzed must be a network-type problem. In other words, network analysis is not appropriate for all situations. Next, network analysis does involve a degree of abstraction for the actual situation. The particular problems must be represented by decision points, events, etc. Varying levels of detail can be used in the definition of the problem as a network. Finally, the subjective estimates describing the events in the network (e.g., cost) suffer from the same problems as subjective estimates as a whole do.

#### 4.3.1.9 Decision Trees.

##### 4.3.1.9.1 Description.

Decision trees are used for the examination of decisions by breaking them into the sequences of supporting decisions and the resulting uncertain occurrences. Figure 4-2 is an example of a decision tree.

In this particular example, the left-hand square is the starting point of the sequence of decisions. The two circles to the right of the square represent either of two ways in which process can move from the initial point. In this case, either a tilt-wing design can be chosen or a helicopter can be chosen. From the circles, three possible outcomes can occur on the top branch (tilt-wing) or two possible outcomes can occur on the bottom branch (helicopter). The likelihood of each of these outcomes is shown on the appropriate branch of the decision tree. These

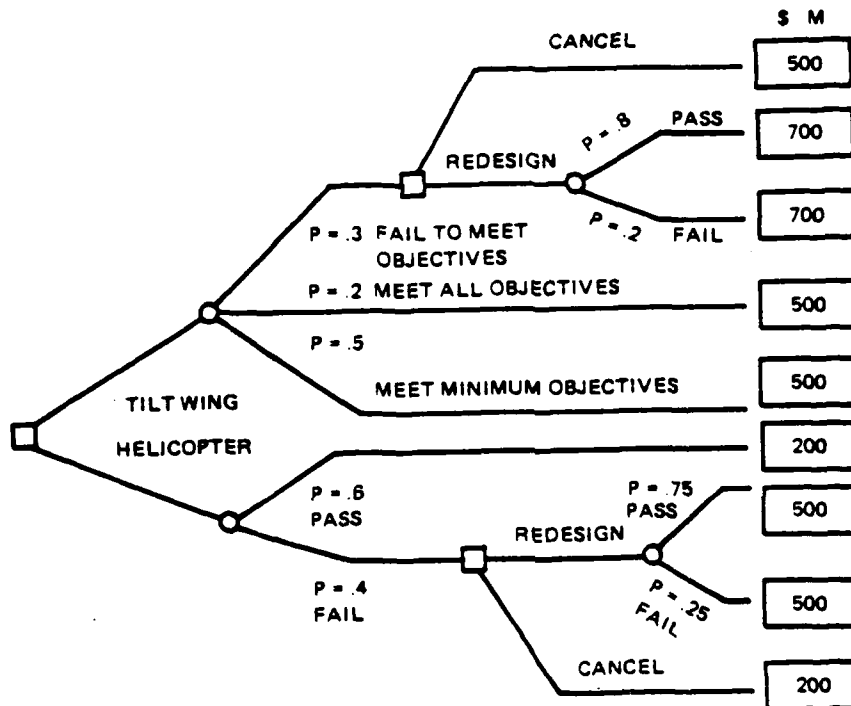


Figure 4-2. Example: Decision Tree Techniques

likelihoods, or probabilities, are assigned by some method of eliciting subjective estimates from a panel of experts. This general process of designing a decision tree is carried out until the final consequences of each possible branch of the tree are shown on the far right-hand side. In this example, cost judgments are shown. Again, these judgments are obtained via some subjective method of getting at expert opinions.

#### 4.3.1.9.2 Advantages.

The decision tree method is an easy and straightforward method of modeling the possible outcomes of a situation. The method can be implemented without extensive mathematical training and often provides a good analysis of the situation.

#### 4.3.1.9.3 Limitations.

Again, as in the case of network analysis, the decision tree is an abstraction of the entire decision process. Thus, first, all of the possible decisions should be known. Second, the probabilities assigned to each branch of the decision tree must be specified. Decision tree analysis presents no formal way of estimating probabilities. Lastly, the final outcomes (e.g., cost values) must be estimated. Again, no explicit method is specified in the decision tree approach. In essence, decision tree analysis is a framework for examining probabilities and uncertainty once these values are estimated.

#### 4.3.2 Objective Risk Techniques.

The probability density function can also be estimated from objective data. Parametric models are used for risk assessment where objective data is available. Where extensive objective data bases exist, accurate risk models have been developed. The insurance industry immediately comes to mind. With a great deal of accuracy, the auto

insurance business can determine the probability of an accident of varying degrees of severity.

#### 4.3.2.1 The Difficulty of Making Objective Estimates.

Making objective estimates, i.e., obtaining relationships among variables based on objectively-derived data, is not necessarily difficult. Whether the relationship correlates to the desired implication is what is difficult to determine. The risk relationships derived (e.g., by regression techniques) will involve only some of the independent defining variables, hence the derived model of reality may not include one or more key risk drivers.

Because of the complexity involved with considering too many risk variables, the usual technique is to determine the key risk drivers. This is often a very difficult task, if not impossible. Also, each particular case may require somewhat different drivers. Objective estimates depend upon the collection of accurate quantitative data, frequently based on ordered, equal interval scale (e.g., numeric, integer). It is frequently very difficult to assign a quantitative value to a variable.

For example, assigning a numeric value between 1 and 10 to software product quality is much more difficult than obtaining the number of alcohol-related automobile fatalities for each of the fifty states. The essence of the difficulty is that "software product quality" has many more defining characteristics than does alcohol-related automobile fatalities. The tendency in making objective estimates is then to decompose the "software product quality" into a more definitive hierarchy of "single-dimension" characteristics. The frequent net result is numerous characteristics, nebulous and inconsistent value scales (e.g., (0,1) scale, percent scale, (no, yes) scale, 1..6 discrete scale), inaccurate variable values, and even more questionable derived relationships.

Of course, subjective risk estimation has many of the same difficulties as objective risk estimation. However, objective estimation may



result in a more dangerous reliance upon "numbers" rather than concept. The tendency is to forget the objective-based theoretical risk foundation (i.e., the basis and meanings of the numbers). This may result in a blind application (perhaps misapplication) of data to crunch out a computer-generated value with little or no evidence of how the value was derived. Thus, it may be known that software supportability is risky, but not why it is.

#### 4.3.2.2 Parametric Models.

Parametric estimating relationships are the result of mathematical methods that determine a relationship between some variable of interest (e.g., cost) and measurable system characteristics such as code length, maturity, documentation, etc. The method makes use of a statistical technique called regression analysis to develop an equation to fit a body of data. The data consists, in this example, of known costs and the associated system characteristics such as code lengths, maturity measures, documentation measures, etc.

Any model is an abstraction of reality, by definition. A model is a way of summarizing, representing, and expressing in a formal way the complex relationships and interrelationships of reality. In this context, reality is the software supportability problem. Thus, it is realized that any parametric model will not account for every detail affecting the dependent variable. Any evaluation of the dependent variable must be accompanied by a caveat of what is included or excluded in the model. Given that a model is an abstraction, then the first objective is to identify the main drivers of the dependent variable. In other words, those components that account for the most variation in the uncertainty in cost, scheduling, or performance of software supportability will be considered first in the model development.

The aim in developing a parametric model is to first keep it fairly parsimonious. The rationale for parsimony is several fold. One, by focusing the model only on the key drivers, or independent variables,

undue complexity is avoided. Increased model complexity may further account for the variation in a given data set, but the applicability of complex models to novel situations can be questionable. Also, as complexity and detail in a model are added, then it is implied that the exactness of the model improves. This may not be the case. Further, following Rowe's (1977) ideas, detailed models may contain more descriptive certainty, yet there is an increased measurement uncertainty. That is, detailed concepts may be included in a model, but the measurement of these concepts quickly becomes problematic. Finally, initial attempts at parametric modeling should not get bogged down in detail. Refining a model comes later after the major pieces of a model are in place. Thus, it is the intent in parametric modeling to first introduce the main drivers.

Any parametric model will not simply estimate some definitive quantity of the dependent variable. Instead, the model must provide, in some way, a set of probabilities. That is, some measure of the variation must be at least appended to the expected value of the dependent measure (e.g., cost). The model must incorporate some notion of the statistical uncertainty of the supportability expense. In this way, the model touches base with the theoretical basis of risk. Some estimate of a probability density function must be predicted, however crude.

As previously stated, the risk assessment model will be a fairly simplistic one. Perhaps only seven or eight risk factors will be modeled to predict the cost, schedule, or performance measures of supportability. Factors such as maintenance and reliability have received considerable attention in terms of attempting to measure these concepts. This previous research may be useful. Preexisting parametric-type relationships can be directly incorporated into a model (given an understanding of their applicability). More often than not, however, well-defined pieces of a model will not exist. For this scenario, the structural relationship of the model must first be determined. For instance, the cost of supportability may be an inverse function of the amount of code documentation. In some cases, the driving factors may not easily be measured.

Then, a proxy variable or a set of proxies will be used. Where data exists matching the structural model, then parametric relations can be developed via regression techniques. Jackknife or bootstrap methods can be used to incorporate uncertainty into the model (see Efron, B., reference 5.28). Where data is sparse or nonexistent, then equations can be developed that are heuristics or "rules of thumb." As an example, higher level computer languages are easier to modify than assembly language codes. This concept may be incorporated into a parametric model as a multiplying factor of some sort. The heuristics can be developed by analogy, from concepts published in the literature, from intuition, or from some reasonable method of obtaining subjective estimates.

Technical issues of the parametric modeling task are also apparent. Of critical importance is the way in which the components or drivers of the model are combined together. Specifically, if a parametric model estimates probability density functions of cost for only two drivers, say maintenance requirements and code characteristics, then it may be problematic in combining the estimates into a total estimate of the dependent variable. The interdependence among drivers causes mathematical complications in building a total probability distribution of the dependent variable. (See Worm's (1981) paper for some ideas in this area.) Another issue is the distributional form of the probability density function describing the dependent variable. Where the probability density function is not completely and entirely determined, then some distributional form is assumed. This assumption makes the modeling process traceable in that only moments (e.g., mean, standard deviation) of the distribution need be estimated. From the risk literature reviewed, normal, beta, triangular, Weibul, and Rayleigh distributions have all been considered.

#### 4.3.3 Decision Theory.

Every day, in our professional work and our personal lives, each of us must make a multitude of decisions. Both major and minor, under

various conditions of uncertainty and partial ignorance. Decision theory deals with the development of methods and techniques that are appropriate for making these decisions in an optimal fashion. In fact, statistics itself is sometimes described as the science of decision-making under uncertainty. Although decision theory may not be tantamount to the entire field of statistics, the importance of decision theory has steadily grown during the past 30 years as virtually all the classical problems of statistical inference, and many new problems as well, have been formulated in decision-theoretic terms.

The mathematical basis of statistical decision theory was developed mainly by Abraham Wald during the 1940s. In many respects, this theory was an outgrowth, and a special case, of the "theory of games" as developed by von Neumann and others during the 1920s and 1930s. The central difference is that in the theory of "zero-sum two-person games," the decision-maker must act against an intelligent opponent whose interests are diametrically opposed to his own, whereas in a statistical decision problem, there is usually no such opponent. For this reason, the theory of "minimax decision rules," which play a central part in the theory of games, play at best a very minor part in modern decision theory.

It is not appropriate to describe decision theory in any depth in this report. However, an overview of some of the more important theoretical concepts will be presented in the following sections. A conceptual example from Georgia Tech research on applying decision theory to software testing will illustrate some of these ideas.

#### 4.3.3.1 Parameters, Decisions, and Consequences.

Consider a problem in which a decision maker (DM) must choose a decision from some class of available decisions, and suppose that the consequences of this decision depend on the unknown value  $\theta$  of some parameter  $\lambda$ . We use the term "parameter" here in a very general sense, to represent any variable or quantity whose value is unknown to the DM,

but is relevant to his or her decision. Some authors refer to  $\lambda$  as the "unknown state of nature" or "state of the world." The set  $\Omega$  of all possible values of  $\lambda$  is called the parameter space.

The set  $D$  of all possible decisions  $d$  that the DM might make in the given problem is called the decision space.

For each value of  $\theta \in \Omega$  and each possible decision  $d \in D$ , let  $\gamma(\theta, d)$  denote the consequence to the DM if he or she chooses decision  $d$  when the parameter has value  $\theta$ . Let  $\psi$  denote the set of all consequences that might result from all possible pairings of  $\theta$  and  $d$ . If  $\lambda$  has a specified probability distribution, then the choice of any particular decision  $d$  will induce a probability distribution of  $\gamma(\lambda, d)$  on the set  $\psi$  of possible consequences. Hence, the DM's choice among the decisions in  $D$  is tantamount to a choice among various probability distributions on the set  $\psi$ .

#### 4.3.3.2 The Utility Function.

The DM will typically have preferences among the consequences in  $\psi$ . In some problems, these consequences might be monetary gains or losses; in others they might be much more complicated and abstract quantities. In general, the DM's preferences among the consequences in  $\psi$  will result in his or her having preferences among the different possible probability distributions on  $\psi$ . In other words, if the DM could have a consequence from  $\psi$  generated by a random process in accordance with some specified probability distribution, he or she would generally have a preference as to which distribution was used.

Now let  $U$  denote a real-valued function on the set  $\psi$ , i.e., a function that assigns a real number to each consequence in  $\psi$ . Also, for any probability distribution  $P$  on the set  $\psi$ , let  $E(U|P)$  denote the expectation of  $U$  with respect to the distribution  $P$ . Then under certain conditions regarding the coherence of the DM's preferences among probability distributions, it can be shown that there exists such a function  $U$  with the following property: for any two distributions,  $P_1$  and  $P_2$ ,  $P_1$  is not preferred to  $P_2$  if and only if  $E(U|P_1) \leq E(U|P_2)$ .

A function  $U$  with this property is called a utility function, and the value that  $U$  assigns to any particular consequence is called the utility of that consequence. The expected utility hypothesis, as we have just described, states that the DM will prefer a probability distribution  $P$  for which  $E(U|P)$  is as large as possible. In other words, the DM will prefer a distribution for which the expected utility of the resulting consequence is a maximum.

It should be noted that there is more than one utility function that could be used in a given problem. If  $U$  is a utility function, then  $V = aU + b$ , where  $a$  and  $b$  are constants ( $a > 0$ ,  $-\infty < b < \infty$ ) is also a utility function. The reason is that for any two distributions  $P_1$  and  $P_2$ ,  $E(U|P_1) \leq E(U|P_2)$  if and only if  $E(V|P_1) \leq E(V|P_2)$ . Hence, both  $U$  and  $V$  represent the DM's preferences equally well. In practice, this arbitrariness is exploited and removed by choosing two particular consequences and assigning them the utilities 0 and 1, or 0 and 100, or some other convenient pair of reference values.

#### 4.3.3.3 Components of a Decision Problem.

We now return to the original decision problem. For each value of  $\theta \in \Omega$  and each decision  $d \in D$ , let  $U(\theta, d)$  denote the utility of the consequence  $\gamma(\theta, d)$ . We may think of  $U(\theta, d)$  as the utility of choosing decision  $d$  when the parameter  $\lambda$  has the value  $\theta$ . Suppose that  $\lambda$  has a specified probability distribution  $\xi$ . Then in accordance with the expected utility hypothesis, the DM will choose a decision  $d$  for which the expected utility  $E(U|\xi, d)$  is a maximum. Such a decision is called an optimal decision or a Bayes decision with respect to the distribution  $\xi$ .

In many decision problems, it has become standard to specify the negative of the utility function, rather than the utility function itself, and to call this function the loss function. Thus the loss  $L(\theta, d)$  is the disutility to the DM of choosing decision  $d$  when the parameter has the value  $\theta$ . An optimal or Bayes decision with respect to the distribution  $\xi$  will be a decision  $d$  for which the expected loss  $E(L|\xi, d)$  is a minimum.

Thus the components of a decision problem are a parameter space  $\Omega$ , a decision space  $D$ , and a loss function  $L(\theta, d)$ . For any given distribution  $\xi$  of  $\lambda$ , the expected loss  $E(L|\xi, d)$  is called the risk  $R(\xi, d)$  of the decision  $d$ . The risk of the Bayes decision, i.e., the minimum  $R_0(\xi)$  of  $R(\xi, d)$  over all decisions  $d \in D$ , is called the Bayes risk.

#### 4.3.3.4 Subjective Probability.

In some decision problems, the probability distribution  $\xi$  that the DM assigns to  $\lambda$  will be based on a large amount of historical data or on theoretical frequency considerations. In such problems, the distribution  $\xi$  will be "objective" in the sense that any other DM who faced the same problem would assign the same distribution. In most decision problems, however, the distribution  $\xi$  will be a "subjective" distribution that is based, at least in part, on the DM's personal information and beliefs about what the value of  $\lambda$  is likely to be.

The existence of subjective probabilities is based on the assumption that certain conditions are satisfied regarding the coherence of the DM's judgments about the relative likelihoods of various subsets of values of  $\lambda$ . When these conditions are satisfied, it can be shown that there exists a unique probability distribution  $P$  on the set  $\Omega$  that satisfies all the mathematical properties of probability and has the additional property that for any two subsets  $A \subset \Omega$  and  $B \subset \Omega$ ,  $P(A) \leq P(B)$  if and only if the DM does not believe that the value of  $\lambda$  is more likely to lie in  $A$  than in  $B$ .

Some statisticians feel that there are different types of probability and that subjective probabilities are of a different type from logical, frequency, or physical probabilities. On the other hand, it can be argued that subjective probability is the only type of probability that can be put on a sound foundation and the only type of probability that exists. In this view, all probabilities are subjective; some are more "objective" than others only because larger groups of DM's would all assign the same values for these probabilities based on their experience.

Together, the concepts of subjective probability and utility provide a unified theory of decision making. The DM's subjective probabilities represent his or her knowledge and beliefs, and the DM's utilities represent his or her tastes and preferences. The expert DM is careful to maintain the distinction between these concepts, and does not confuse the value that he or she wishes  $\lambda$  would have with the value that he or she thinks  $\lambda$  is likely to have. In other words, the DM does not let utilities influence his or her subjective assignment of probabilities, and vice versa. The DM then chooses a decision that maximizes his or her subjective expected utility or, equivalently, minimizes his or her subjective expected loss.

#### 4.3.3.5 Decision Analysis.

Many problems of decision making, such as deciding where to locate a new airport, are extremely complicated, and it is often not immediately clear how to apply the concepts of decision theory that have just been described. The process of aiding the DM in applying these concepts in a particular problem is called decision analysis. In recent years techniques of decision analysis have been developed which are intended to aid the DM in (a) identifying all the relevant dimensions of the parameter  $\lambda$ , (b) specifying the spaces  $\Omega$  and  $D$  of all possible parameter values  $\theta$  and decisions  $d$ , and especially (c) specifying the DM's probabilities and utilities.

Various procedures are available, including some computer programs, for the elicitation of a DM's subjective probabilities. A probability distribution on  $\Omega$  must be determined on the basis of the DM's responses when questioned about the relative likelihoods of different events. Some type of fitting procedure is typically needed because few, if any, persons exhibit the perfect coherence necessary for the existence of a unique distribution. Similarly, procedures are available for fitting a utility function on the basis of the DM's responses when questioned about his or her preferences among different probability distributions that might yield a consequence from the set  $\psi$ .



#### 4.3.3.6 Tests of Hypothesis.

The standard problems of testing hypotheses can also be formulated as decision problems. In fact, every test of hypothesis is, at least theoretically, a problem with exactly two decisions: accept the null hypothesis  $H_0$ , which we shall call decision  $d_0$ , and accept the alternative hypothesis  $H_1$  (or, equivalently, reject  $H_0$ ), which we shall call decision  $d_1$ .

Loss functions appropriate to testing hypotheses can easily be developed. For example, suppose that  $\lambda$  is a real-valued parameter and it is desired to test the hypotheses  $H_0: \lambda \leq \theta_0$  and  $H_1: \lambda > \theta_0$  where  $\theta_0$  is a specified number. A typical loss function for this problem would have the following form:

$$\begin{array}{ll} L(\theta, d_0) = 0 & \text{for } \theta \leq \theta_0, \\ L(\theta, d_0) = \alpha_0(\theta) & \text{for } \theta > \theta_0, \\ L(\theta, d_1) = \alpha_1(\theta) & \text{for } \theta \leq \theta_0, \\ L(\theta, d_1) = 0 & \text{for } \theta > \theta_0. \end{array}$$

where  $\alpha_0(\theta)$  is positive and nondecreasing for  $\theta > \theta_0$  and  $\alpha_1(\theta)$  is positive and nonincreasing for  $\theta < \theta_0$ . The posterior PDF of  $\lambda$  can be calculated from any specified prior PDF. The Bayes test procedure would then choose the decision with the smaller posterior risk.

#### 4.4 APPLICATION MODELS.

The following sections describe two actual models which have been proposed for risk assessment of software supportability. The first model described is currently being developed at Georgia Tech. The second model was proposed by Fisk and Murch (reference 5.12). Both efforts to model risk for software supportability represent the only models that appear to exist for this particular problem.

Neither the Georgia Tech model nor the proposed Fisk/Murch model have been identified with using either subjective or objective data. Apparently, both models could incorporate either or both types of data.

Subjective data are those in which no well-defined rules are used to assign a value (usually a number) to describe some characteristic. For instance, verbal estimates of code complexity are subjective data. On the other hand, if the complexity of a piece of code is estimated as "high" because there are more than twenty modules, then the data is objective. In this instance, a well-defined rule existed which estimated complexity based on the number of modules.

#### 4.4.1 Georgia Tech Conceptual Model: Software Testing.

Georgia Tech personnel (reference 5.39) are in the conceptual phase of developing a risk model for software testing. This model is essentially a top down approach based upon decision theory. The briefing slides of reference 5.39 are not intended to be an in-depth analysis and any conclusions are premature at this time, but the top view of the model does illustrate some aspects of decision theory such as the "utility" function.

##### 4.4.1.1 Description.

The basic information desired from a risk model on software testing is the selection of tests based upon optimization of residual risk, and the determination of when it is more costly to continue testing than the residual risk warrants.

The tester is presented with a set of possible tests ( $T_i$ :  $i = 1..n$ ) and a set of test strategies ( $A_i$ :  $i = 1..m$ ) where each  $A_i$  may be one or more test,  $T_j$  in some test sequence based upon a desired strategy (e.g., highest reliability possible). When adopting any particular test strategy, a set of consequence events can be observed (e.g., hardware device  $x$  emits incorrect data value  $y$ ). This set of possible observed events while the system is operating is denoted ( $S_i$ :  $i = 1..e$ ). For each possible pair ( $A_i, S_j$ ) a utility (tester's) value  $U_{ij}$  is determined and a regret (in not applying another strategy) value  $r_{ij}$  is determined. The

goal is to rank the tester's choices with respect to the  $U_{ij}$ 's,  $r_{ij}$ 's, and various optimality criteria determined by the test strategy.

The following test policies are illustrated:

- a) Policy 1. High cost test should be applied to reduce risk of critical system error.
- b) Policy 2. Apply least cost test and justify incremental costs in future tests.
- c) Policy 3. Same as Policy 2 for cases where cost of test may not be linearly ordered (i.e., parallel paths may be alternative choices).

Ruby's Theory of Test Utility is briefly presented. It is based upon test definition dependencies: number of variables, number of domains, structural complexity. The measure of test cost is based upon the difference in the test definition dependencies from one test to another. The utility function for a given test is then the product of this "difference" and the cost incurred due to any remaining errors not determined by the given test. Of course, all these dependencies, differences, and potential cost incurred by residual errors, are usually very difficult to measure or estimate. Some possible methods of determining the utility function are illustrated in probability linguistics and possible test strategies described.

Some axiomatics regarding the software test risk assessment are given:

- a) Completely order test strategies
- b) Isolate optimal set.

Some guidance and terminology is presented for doing this. To quantify the risk, it is suggested that utility functions be derived using a conjecture by Ruby concerning a simplification of the utility function equation, that the differences in the derived utility values be demonstrated to be a measure of risk, and that the preference relations for testing be classified.

#### 4.4.1.2 Advantages, Limitations.

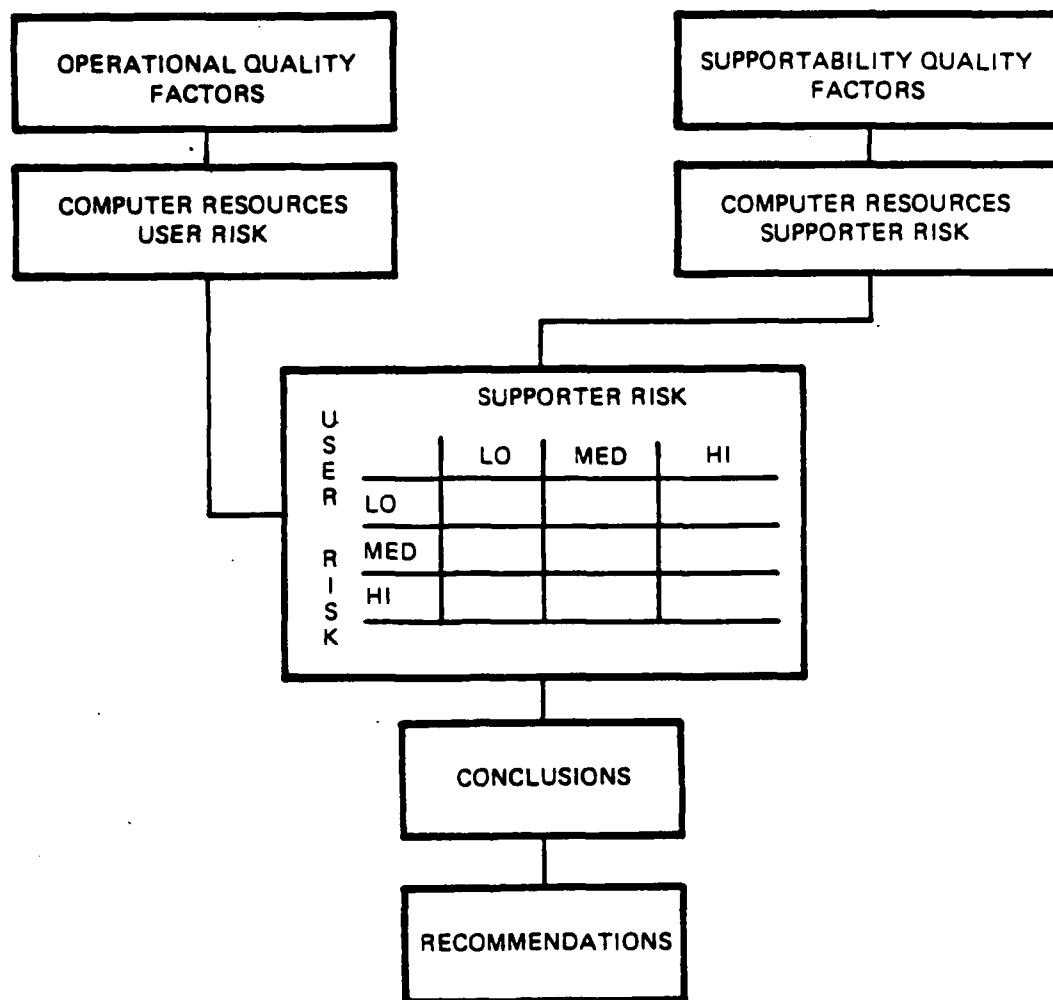
The information in reference 5.39 is at best sketchy and the interpretation applied in this report is probably not totally accurate. The suggested model needs a great deal of refinement and a more practical application example, even to be reasonably understood. However, it does appear that this type of model, if it existed in some reasonably validated form, would have an importance for an AFOTEC RAMSS. The test completeness index described in reference 5.12 is dependent upon a test strategy. Knowing the various risks associated with the test strategies would first allow AFOTEC to better utilize test resources against expected risk reduction and second provide test completeness risk MOEs of more substantial meaning than the index suggested in reference 5.12.

#### 4.4.2 Proposed Fisk/Murch Model.

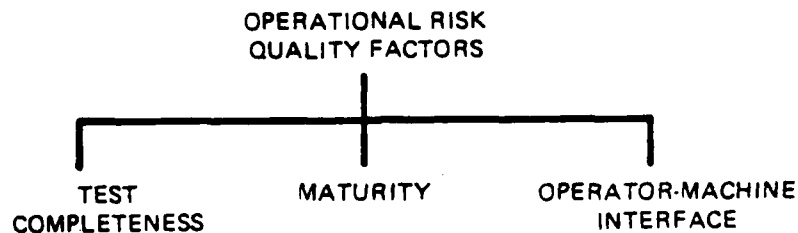
AFOTEC prepared a proposal for computer resources risk assessment during OT&E (reference 5.12) as justification for this feasibility study. This effort was the only literature reviewed which directly addressed the integration of risk assessment and software supportability. The proposal is preliminary and is not officially sanctioned by AFOTEC. It was not meant to be classified as a methodology or technique, but was simply meant to be used as a guide to further study. Its importance derives from the practical view presented of evaluating and reporting software user and supporter risks associated with acceptance of computer resources, especially software.

##### 4.4.2.1 Description.

This proposal presents a framework for software risk assessment. This framework integrates aspects of current AFOTEC developed methodologies for evaluating computer resources as part of OT&E activities without restricting the possibility of including other methodologies. The structure of this framework is shown in figure 4-3.



(a) Framework for Computer Resources Risk Assessment



(b) OT&E Software Operational Quality Factors

Figure 4-3. Proposed Fisk/Murch Framework for Computer Resources Risk Assessment

The technique of closed form questionnaires (section 4.3.1.6) is used to arrive at software supportability risk "quality factor" ratings, and software operational risk "quality factor" ratings. These ratings are then normalized to a value between 0 and 1. The independent quality factor risk is simply defined to be one minus the normalized rating. Thus, the higher the evaluated quality factor value, the lower the quality factor operational or support risk and vice versa.

Because quality factors are not independent and usually not of equal importance, this risk assessment technique allows for a factor influence matrix and a quality factor relative importance weight. The analytical procedure is illustrated in figure 4-4. The procedure is applied for user and for supporter (the risk agents in this method). The resulting risk values lie between 0 and 1 for user and supporter. The proposed evaluation criteria for "low," "medium," and "high" risk, and a suggested matrix form representing the results, is shown in figure 4-5.

An example using the test factors of figure 4-3 for user and supporter and hypothesized values from an AFOTEC evaluation is presented to illustrate the analytical procedure and the resulting risk matrix. A condensed version of this is shown in figure 4-6.

#### 4.4.2.2 Advantages.

The advantages of the proposed Fisk/Murch model are primarily due to its direct applicability and simplicity. This model can easily be applied within AFOTEC evaluation constraints (resources, time). It is generic in that other quality factors can be easily added or the current ones modified. The model is not dependent on how the risk MOE is actually computed. Thus different algorithms than the ones proposed could be used. Or perhaps one of the subjective or objective techniques discussed in sections 4.3.1 or 4.3.2 could be used in combination with the current AFOTEC closed form questionnaire evaluations to estimate the risk MOE in a more probabilistic-based manner.

The model does provide a quick pointer hierarchy to potential problem (risky) areas, from the decision maker risk matrix to the user/supporter risk agent, to the risk agent quality factors, to the quality

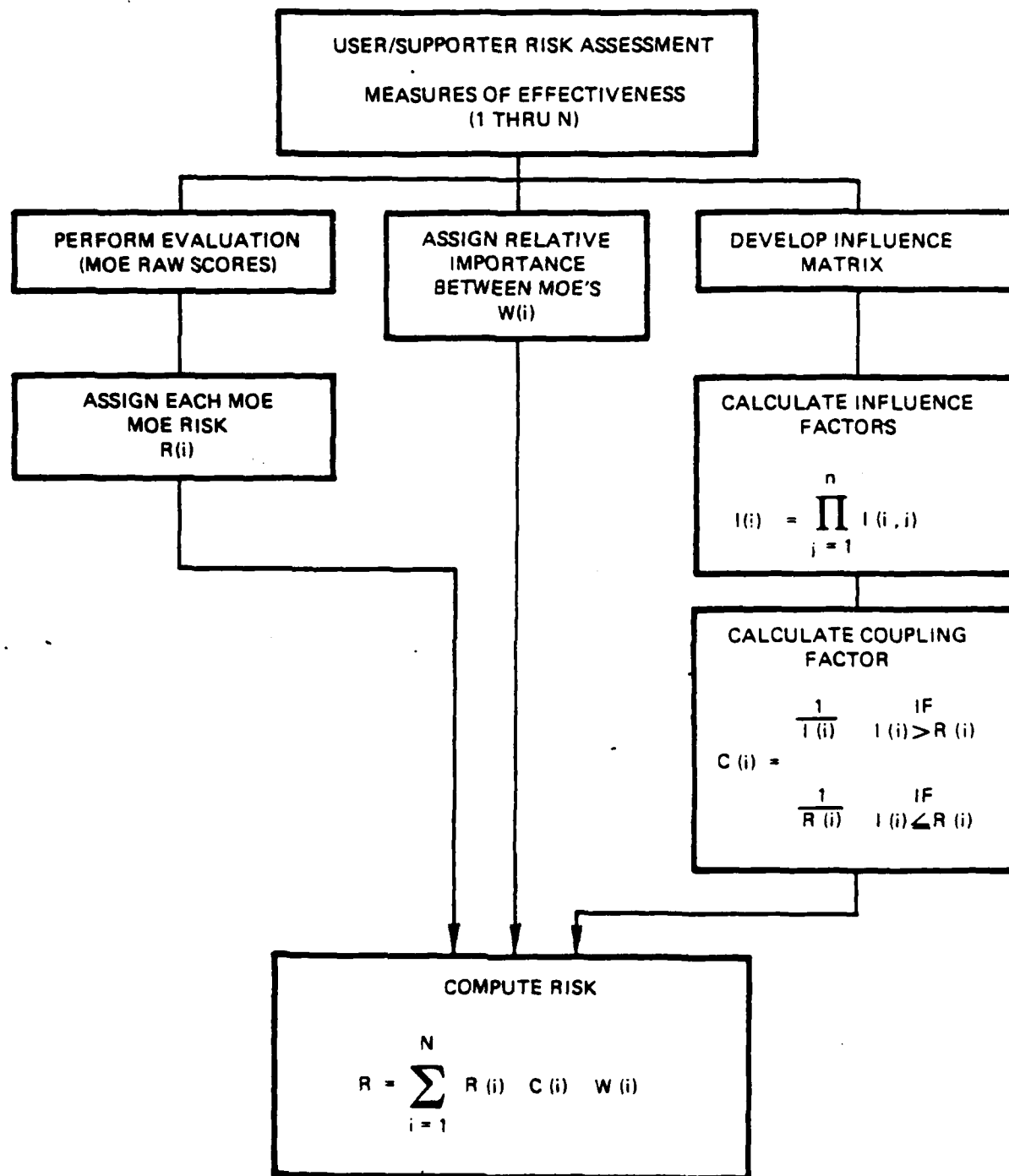


Figure 4-4. Proposed Fisk/Murch Risk Assessment Analytical Procedure

RISK MATRIX			
SUPPORTER RISK USER RISK	LO	MED	HI
LO			
MED			
HI			

LO = LOW RISK (0.00 - 0.17)  
MED = MEDIUM RISK (0.18 - 0.44)  
HI = HIGH RISK (0.45 - 1.00)

Figure 4-5. Proposed Fisk/Murch Evaluation Criteria and Risk Matrix



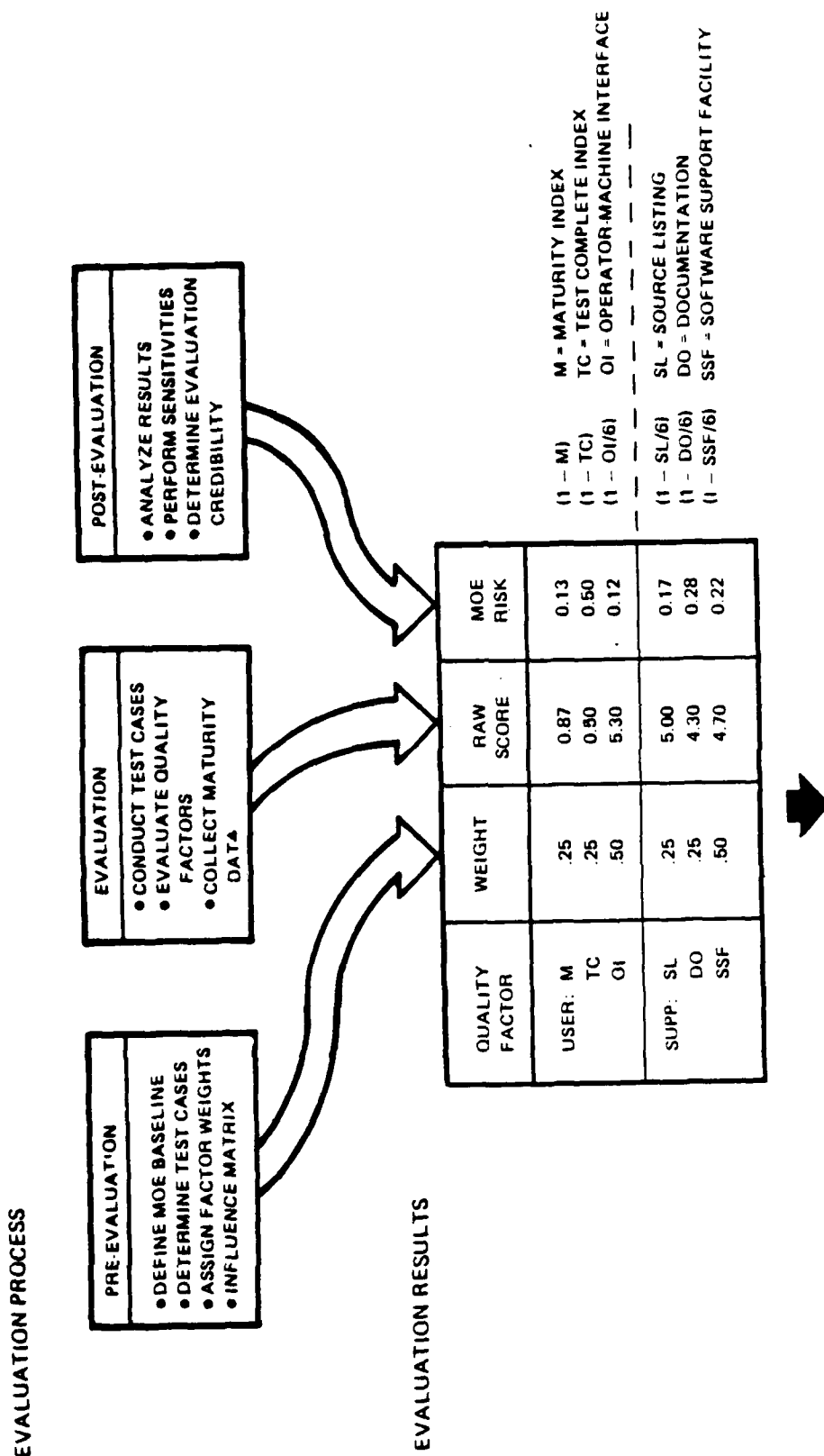


Figure 4-6. Example: Proposed Fisk/Murch Model

ANALYTIC PROCEDURE  
(APPLY INFLUENCE MATRIX)

USER

$I(i,j)$				$I(i)$				$C(i)$				RISK
QF	M	TC	O	M	TC	OI		M	TC	OI		
	1	.5	1					2.00				.30
M				0.50				1.00				
TC	1	1	1		1.00				1.00			
OI	.87	.5	1		.43			2.33				

SUPPORTER

QF	SL	DO	SSF	SL	DO	SSF	RISK
	1	1	1	1	1	1	
SL							.15
DO	1	1	1				
SSF	1	1	1	1			

DECISION MAKER RISK MATRIX

SUPP RISK USER RISK	LO	MED	HI
LO			
MED			
HI			

$$\begin{aligned} \text{SUPPORTER RISK} &= \text{LO} (0.00 \cdot 0.17) \\ \text{USER RISK} &= \text{MED} (0.18 \cdot 0.45) \end{aligned}$$

Figure 4-6. Example: Proposed Risk/Murch Model (Concluded)

test factors, and eventually to the individual test factor characteristics. This is very comforting in as subjective an area as software evaluation to be able to point to lower level details to support evaluation conclusions.

#### 4.4.2.3 Limitations.

The most severe limitation of the proposed Fisk/Murch model is its lack of a theoretical risk foundation. The risk MOEs have no direct connection to "probability of a negative outcome." The reason for this is that the model is not connected to actual support activity and consequences other than the quality factors which are supposed to represent such activity and consequences. The use of subjective weights as importance factors has the usual limitation and constraint, i.e., the weights are based on "gut feel." It is very easy to change a computed risk by one category (LO-MED, MED-HI) by manipulating the weight.

The specific framework does include the concept of user and supporter as risk agents. The use of the influence matrix recognizes the potential dependent interaction of what one would like to design as independent quality factors. However, there is no recognition till the final integration of risk into the decision-maker matrix, that the user and supporter have interdependencies. For example, the turn-around time required by the user as part of the "emergency maintenance request" will dictate the evaluation results for a software support facility (a supporter factor). Thus, as the user risk from support service decreases (i.e., turnaround time is reduced), the corresponding supporter risk increases. The manner in which influence matrix values are computed is also highly questionable from a feasibility viewpoint. It is probably impossible to determine an accurate numerical value for the impact of test completeness upon maturity or vice versa. The non-symmetric nature of this relationship was alluded to by the form of the user influence matrix in figure 4-6, but the proposal did not make it clear.

The combination of scales with vastly different meanings (e.g. test completeness and operator-machine interface) is a severe limitation for

the user quality factors. Note that the scales for the supporter quality factors are the same. Also, there are some minor technical inconsistencies with the normalization of values. For example the MOE risk for source listings is  $(1 - SL/6)$ , but the source listing values (SL) range from 1 to 6, so a risk value of 1 is impossible. Perhaps a better normalization form would be  $(1 - (SL-1)/5)$ .

**Section V**  
**References**

SECTION V  
REFERENCES

- 5.0 "Software Risk Assessment in OT&E," Final Subtask Statement 304 for AFOTEC Contract F29601-80-C-0035, AFOTEC, Kirtland AFB, NM, Apr 84.
- 5.1 AFOTEC 800-2 Volumes 1 through 7, Software OT&E Guidelines.
- 5.2 FIPS PUB 31, "Guidelines for ADP Physical Security and Risk Management," National Bureau of Standards, Jun 74.
- 5.3 FIPS PUB 65, "Guidelines for Automatic Data Processing Risk Analysis," National Bureau of Standards, Aug 79.
- 5.4 AFR 205-16, "Automatic Data Processing (ADP) Security Policy, Procedures, and Responsibilities," Attachment 5: Guidance for Performing Risk Analysis, 1 Aug/84.
- 5.5 OPNAVINST 5239.1A, "Department of the Navy Automatic Data Processing Security Program, Appendix E: Risk Assessment Methodology, 3 Aug 82.
- 5.6 Lathrop, F., "Alternative Methods for Risk Analysis: A Feasibility Study," Air Force Computer Security Program Office, 1 Sep 81.
- 5.7 Neugent, W., "Technology Assessment: Methods for Measuring the Level of Computer Security," Section 4.2: Risk Assessment Methodologies, National Bureau of Standards, Draft, Sep 81.
- 5.8 RADC, "Reliability Model Demonstration Study," RADC-TR-83-207, Volumes I and II, Aug 83.
- 5.9 Directorate of Aerospace Safety, "A Risk Management Guide for Air Force Operations," Air Force Inspection and Safety Center, Norton AFB, CA, 6 Nov 79.
- 5.10 USAF Scientific Advisory Board, "The High Cost and Risk of Mission-Critical Software," USAF SAB Ad Hoc Committee, Dec 83.
- 5.11 Fisher, G. and Lt. Col. E. Gay, "An Approach to Risk Analysis: A Process View," AF/SA Technical Note, Jun 81.
- 5.12 Fisk, F., and W. Murch, "A Proposal for Computer Resources Risk Assessment During Operational Test and Evaluation," AFOTEC Draft Report, 3 Oct 83.

- 5.13 Peercy, D., "A Framework for Software Maintenance Management Measures," Proceedings of the Seventeenth Annual Hawaii International Conference on System Sciences, Jan 84.
- 5.14 Peercy, D., and G. Swinson, "A Software Support Facility Evaluation Methodology," Proceedings of Symposium on Application and Assessment of Automated Tools for Software Development, Nov 83.
- 5.15 Booch, G., Software Engineering with Ada, Reading, MA: Benjamin/Cummings, 1983.
- 5.16 LeBlanc, R., and J. Goda, "Ada and Software Development Support: A New Concept in Language Design," Computer, 15(1982), 5, pp. 75-82.
- 5.17 Howden, W., "Contemporary Software Development Environments," Communications of the ACM, 25(1982), 5, pp. 318-329.
- 5.18 Lientz, B., and E. Swanson, Software Maintenance Management, Reading, MA: Addison-Wesley, 1980.
- 5.19 Lientz, B., and E. Swanson, "Problems in Application Software Maintenance," Communications of the ACM, 24(1981), 11, pp. 763-769.
- 5.20 GAO Report, "Federal Agencies Maintenance of Computer Programs: Expensive and Undermanaged," AFMD-81-25, Feb 81.
- 5.21 Parikh, G., Techniques of Program and System Maintenance, Cambridge, MA: Winthrop, 1982.
- 5.22 Thayer, R., A. Pyster, and R. Wood, "Validating Solutions to Major Problems in Software Engineering Project Management," Computer 15(1982), 8, pp. 65-77.
- 5.23 Boehm, B., J. Brown, and M. Lipow, "Quantitative Evaluation of Software Quality," Proceedings 2nd International Conference on Software Engineering, San Francisco, CA: 1976, pp. 592-605.
- 5.24 McCall, J., and M. Matsumoto, "Software Quality Measurement Manual," RADC-TR-80-109, Vol II (of two), Apr 80.
- 5.25 Rowe, W., An Anatomy of Risk, J. Wiley and Sons, New York, 1977.
- 5.26 Atzinger, E., and Brooks, W. (eds), "A Compendium on Risk Analysis Techniques," Aberdeen Proving Grounds, U.S. Army Materiel Systems Analysis Agency, 1972.
- 5.27 Megill, R., An Introduction to Risk Analysis, Petroleum Publishing, Tulsa, 1977.

- 5.28 Efron, B., The Jackknife, Bootstrap and Other Resampling Plans, Society for Industrial Mathematics, Philadelphia, 1982.
- 5.29 Worm, G., "Applied Risk Analysis with Dependence Among Cost Components," Clemson University Department of Industrial Management, 1981.
- 5.30 Rescher, N., Risk, Washington, D.C.: University Press of America, 1983.
- 5.31 Hoessel, W., W. Huebner, D. Peercy, G. Richardson, "Software Supportability Risk Assessment in OT&E: Literature Review, Current Research Review, and Data Base Assemblage," BDM/A-84-322-TR (Draft), The BDM Corporation, July 1984.
- 5.32 "Analysis Support for Computer System Security OT&E," Final Subtask Statement 294 for AFOTEC Contract F29601-80-C-0035, AFOTEC, Kirtland AFB, NM, Jan. 1984.
- 5.33 Leibowitz, S., S. Parratto, D. Peercy, H. Pringle, J. Wiley, E. Witzke, "Computer System Security (CSS) Literature Review, Current Research Review, and Data Base Assemblage," BDM/A-84-108-TR (Interim), The BDM Corporation, May 1984.
- 5.34 Parratto, S., D. Peercy, H. Pringle, "Computer System Security (CSS) Test and Evaluation (T&E) Life-Cycle Process Definition," BDM/A-84-320-TR (Draft), The BDM Corporation, July 1984.
- 5.35 Defense Systems Management College, Risk Assessment Techniques, Fort Belvoir, Virginia, July 1983.
- 5.36 Apostolakis, G., "Bayesian Methods in Risk Assessment," Advances in Nuclear Science and Technology, New York: Plenum, 1981.
- 5.37 Behm, R. D., J. W. Vaupel, Quick Analysis for Busy Decision Makers, New York: Basic Book, 1982.
- 5.38 Pritsker, A. A. B., C. D. Regden, Introduction to SLAM and Simulation, New York: John Wiley, 1979.
- 5.39 DeMillo, R., "A Risk Model for Software Testing," Georgia Tech University, Briefing Slides, July 20, 1984.
- 5.40 AFOTECR 55-1(C2), "OT&E Reporting," Chapter 6, March 15, 1984.
- 5.41 Peercy, D., "A Software Maintainability Evaluation Methodology," Transactions on Software Engineering, Vol. 7, No. 4, July 1981.



## **Appendix A**

### **Acronyms**

APPENDIX A  
ACRONYMS

ACM	Association for Computing Machinery
ADP	Automatic Data Processing
ADPE	Automatic Data Processing Equipment
ADPF	Automatic (Automated) Data Processing System
ADPS	Automated Data Processing System
ADS	Automated Data System
AFOTEC	Air Force Operational Test and Evaluation Center
AFR	Air Force Regulation
APSE	Ada Programming Support Environment
CCB	Configuration Control Board
CDR	Critical Design Review
CER	Cost Estimating Relationship
CI	Configuration Item
CM	Configuration Management
CMP	Configuration Management Plan
CMS	Configuration Management System
CPCI	Computer Program Configuration Item
CPU	Central Processing Unit
CRISP	Computer Resources Integrated Support Plan
CSS	Computer System Security
DAA	Designated Approving Authority
DBCR	Data Base Change Request
DCP	Decision Coordinating Papers
DID	Data Item Description
DM	Decision Maker
DPI	Data Processing Installation
DoD	Department of Defense
DPESO	DoD Product Engineering Services Office
DSARC	Defense System Acquisition Review Council
DTIC	Defense Technical Information Center

ECS	Embedded Computer System
ELC	Emergency Low Complexity
FCA	Functional Configuration Audit
FIPS	Federal Information Processing Standard
GAO	Government Accounting Office
ICA	Independent Cost Analysis
IOT&E	Initial Operational Test and Evaluation
ISR	Independent Schedule Review
IV&V	Independent Verification and Validation
MAJCOM	Major Command
MOE	Measure of Effectiveness
NBS	National Bureau of Standards
NTIS	National Technical Information Service
OMB	Office of Management and Budget
OPR	Office of Primary Responsibility
OT&E	Operational Test and Evaluation
PCA	Physical Configuration Audit
PDF	Probability Density Function
PDR	Preliminary Design Review
PERT	Program Evaluation and Review Technique
PMD	Program Management Directive
PMP	Program Management Plan
PRR	Program Readiness Review
PVR	Product Verification Review
QA	Quality Assurance
RA	Risk Assessment
RADC	Rome Air Development Center
RAMSS	Risk Assessment Model for Software Supportability
SA	Security Audit
SAB	Scientific Advisory Board
SCP	System Concept Papers

SDR	System Design Requirement
SOA	Special Operating Agency
SS	Software Supportability
SSA	Source Selection Authority
SSAC	Source Selection Advisory Council
SSF	Software Support Facility
SVR	System Validation Review
SWM	Software Maintainability
T&E	Test and Evaluation

## **Appendix B**

### **Glossary**

## APPENDIX B

### GLOSSARY OF TERMS

#### B.1 INTRODUCTION.

The glossary of terms for the Analysis of Software Supportability Risk Assessment models will vary as the project progresses. Refer to BDM/A-84-322-TR, Final to be dated September 28, 1984, for the complete glossary of terms.

Some terms have more than one description; when this is the case, the descriptions either:

- a) Are significantly different between sources (though the effective meaning may be not much different).
- b) Are used differently (different users or technical language).
- c) May be found within the context of a different source.
- d) Have real differences in meaning.

Both DoD and non-DoD (e.g., FIPS PUBs, NBS Special Publications) sources are used. The non-DoD sources and terms are not mandated for our use, but are rather included for breadth of understanding, for those relevant terms commonly used within the non-DoD governmental and/or private sectors.

The source of each description is indicated by a symbol in parenthesis before that source's term description:

```

TERM1
  (SYMBOL1.1)
  Description1.1...
  (SYMBOL1.2)
  Description1.2...
  .
  .
  (SYMBOL1.n)
  Description1.n...
TERM2
  .
  .
  .
TERMN

```

The symbols used and corresponding sources are:

- (AFOTEC1) AFOTEC 800-2, Volume 1, 10 Nov 82, "Software Test Manager's Guide."
- (AFOTEC3) AFOTEC 800-2, Volume III, 1 Jan 84, "Software Maintainability Evaluator's Guide."
- (AFR800-14) Air Force Regulation 800-14, Volume I, "Management of Computer Resources in Systems," 12 Sep 75.
- (AFR300-15) Air Force Regulation 300-15, "Automated Data System Project Management," Jan 78.
- (AFOTEC5) AFOTEC 800-2, Volume 5, 25 Jul 83, "Software Support Facility Evaluation--User's Guide."
- (ROWE) Rowe, William, An Anatomy of Risk, John Wiley, 1977.
- (LATHROP) Lathrop, Frank, "Alternative Methods for Risk Analysis: A Feasibility Study," Air Force Computer Security Program Office, 1 Sep 81.
- (AFR205X) Air Force Regulation 205-16, "Automatic Data Processing (ADP) Security Policy, Procedures and Responsibilities, 1 Aug 84.
- (CURRENT) Current document definition.

## B.2 GLOSSARY OF TERMS FOR THE ANALYSIS FOR DETERMINING FEASIBILITY OF DEVELOPING AND IMPLEMENTING A RISK ASSESSMENT MODEL FOR SOFTWARE SUPPORTABILITY.

### Accuracy

(ROWE)

The quality of being free from error. The degree of accuracy is a measure of the uncertainty in identifying the true measure of a quantity at the level of precision of the scale used for the quantity.

### Algorithm

(AFOTTECP3)

A prescribed set of well-defined rules or processes for the solution of a problem in a finite number of steps.

### Allocated Baseline

(AFR300-15)

The initial approved allocated configuration identification established at end of the definition phase.

### Alternative

(ROWE)

One member of a set of options associated with a decision, the decision being limited to a choice of one and only one.

### Application Functions

(AFOTTECP3)

Any functions which provide specific operational (mission) computations.

### Application Software

(AFOTTECP5)

The software written by software support personnel, or purchased from a contractor, used directly in supporting ECSs. It is normally used for simulation, testing, and ECS code development.

### Application Software (functional)

(AFR205X)

Those routines and programs designed by or for automatic data processing system users and customers to complete specific, mission-oriented task, jobs, or functions, using available automated data



processing equipment and basic software. Application Software may be either general purpose packages, such as demand deposit accounting, payroll, machine tool control, etc., or specific application programs tailored to complete a single or limited number of user functions (for example, base level personnel, depot maintenance, aircraft, missile or satellite tracking, command and control, etc.). Except for general purpose packages that are acquired directly from software vendors or from the original equipment manufacturers, this type of software is generally developed by the user, either with in-house resources or through contract services.

#### Approval to Operate

(AFR205X)

Represents concurrence by the designated approving authority (DAA) that a satisfactory level of security (that is, minimum requirements are met and an acceptable level of risk exists) has been provided, and authorizes the operation of an automated data processing system (ADPS) or network at an automatic data processing facility (ADPF). Approval results from an analysis of the ADPF, ADPS, and automatic data system (ADS) certifications and the operational environment of the automatic data processing (ADP) entity by the DAA.

#### Attributes

(AFOTEC3)

Type, units, range, description, etc., as appropriate.

#### Automated Decisionmaking System

(AFR205X)

Those computer applications which issue checks, requisition supplies, or perform similar functions based on programmed criteria, with little human intervention.

#### Automated Software Development Tool

(AFOTEC5)

A component of System Software that assists in the design, implementation, documentation, and verification of ECS software.

#### Automatic Data Processing Facility (ADPF)

(AFR205X)

The physical resources, including structures or parts of structures, which house and support data processing capabilities. For each computer facility designated as a data processing installation (DPI, reference AFR 300-6), the ADPF is the DPI. For small computers, stand-alone systems, and word processing equipment, the ADPF is the physical area in which the computer is used.

### Automatic Data Processing Resources

(AFR205X)

The totality of automatic data processing equipment, software, data, computer time, computer programs, automatic data processing (ADP) contractual services, ADP personnel, and supplies.

### Availability

(AFR800-14)

A measure of the degree to which an item is in the operable and committable state at the start of the mission, when the mission is called for at an unknown (random) point in time. (MIL-STD-721)

(AFOTEC P5)

The probability that a system is operating satisfactorily at any point in time when used under stated conditions.

### Baseline

(AFR300-15)

A configuration identification document or set of such documents formally designated and fixed at a specific time during a CPC's life cycle. Baselines, plus approved changes to those baselines constitute the current configuration identification.

(ROWE)

A known reference used as a guide for further development activities.

### Bayesian Statistics

(ROWE)

"Bayes rule" (Thomas Bayes, a nineteenth century English mathematician and clergyman) states that the probability that both of two events will occur is the probability of the first multiplied by the probability that if the first has occurred, the second will also occur. Bayesian statistics is a way of making quantity of information substitute for quality of information. There are two kinds of probability: the classical type derived from empirical information, and subjective probability. Bayesian statistics is based on these "subjective probabilities." It involves the joint probability of A and B. The probability of the second event occurring if the first has occurred is called the conditional probability of the second, given the first. Stated another way, the probability of any event  $P(A)$  is always positive but never greater than 1. Symbolically,  $0 \leq P(A) \leq 1$ . If  $P(A) = 0$ , the occurrence of the event B is considered impossible. If  $P(A) = 1$ , the occurrence of the event B is considered to occur with  $P(B)$ .

Benefit

(ROWE)

- a) An axiological concept representing anything received that causes a net improvement to accrue to the recipient.
- b) A result of a specific action that constitutes an increase in the production possibilities or welfare level of society.

Benefit-Cost Ratio

(ROWE)

The ratio of total social benefit to total social costs related to a specific activity.

Capability

(ROWE)

A measure of the degree to which a system is able to satisfy its performance objectives.

Cardinal (interval) Scale

(ROWE)

A continuous scale between two end points, neither of which is necessarily fixed.

Computer Program

(AFR800-14)

A series of instructions or statements in a form acceptable to an electronic computer, designed to cause the computer to execute an operation or operations.

Computer Resources

(AFR800-14)

The totality of computer equipment, computer programs, associated documentation, contractual services, personnel and supplies.

Configuration Control

(AFR300-15)

The systematic evaluation, coordination, approval or disapproval, and implementation of approved changes in the configuration of a CPI after formal establishment of its configuration identification.

Configuration Item (CI)

(AFR300-15)

An item of ADPE that is designated for configuration management.

(AFR800-14)

An aggregation of equipment/software, or any of its discrete portions, which satisfies an end use function and is designated by the Government for configuration management. CIs may vary widely in complexity, size and type, from an aircraft or electronic system to a test meter or round of ammunition. During development and initial production, CIs are only those specification items that are referenced directly in a contract (or an equivalent in-house agreement). During the operation and maintenance period, any reparable item designated for separate procurement is a configuration item.  
(AFR 65-3)

#### Configuration Management (CM)

(AFR300-15)

A management discipline that applies technical and administrative direction and surveillance to:

- (1) Identify and document the functional and physical characteristics of a configuration item.
- (2) Control changes to those characteristics.
- (3) Record and report configuration status.

#### Configuration Management Plan (CMP)

(AFR300-15)

A document which describes project responsibilities and procedures for implementing CM.

#### Configuration Management System (CMS)

(AFOTECPS)

A system applying technical and administrative direction and surveillance to identify and document the functional and physical characteristics of a configuration item; to control changes to those characteristics and to record and report change Processing and implementation status.

#### Consequence Value

(ROWE)

The importance a risk agent subjectively attaches to the undesirability of a specific risk consequence.

#### Consensus

(ROWE)

Group solidarity in sentiment and belief...general agreement.

## Cost

(ROWE)

A result of a specific action that constitutes a decrease in the production possibilities or welfare level of society. Also see Loss.

## Cost-Benefit Analysis

(ROWE)

An attempt to delineate and compare in terms of society as a whole the significant effects, both positive and negative, of a specific action. Generally a number of alternative actions are analyzed resulting in the selection of the alternative that provides either the largest benefit-cost ratio (total benefit/total cost) or one with a positive ratio at least. If an alternative results in a net benefit less than zero or a benefit-cost ratio less than 1, it is deemed socially inefficient and is not carried out.

## Cost-Effectiveness Analysis

(ROWE)

A term less specific than cost-benefit analysis, usually meaning the selection of the lowest cost alternative that achieves a predetermined level of benefits. Alternatively, the analysis and selection of the path that yields the largest social benefit for a predetermined specified level of social costs.

## Critical Automatic Data Processing Resources

(AFR205X)

Those resources that must be protected because their compromise, alternation, destruction, loss, or failure to meet objectives will jeopardize the accomplishment of an Air Force, Air Force subelement, or other service mission or the accomplishment of DoD life support functions.

## Critical Design Review (CDR)

(AFR300-15)

A formal review conducted during the development phase before translating logic, and algorithms to coded instructions.

## Critical Issues

(AFOTECPI)

Those aspects of a system's capability, either operational, technical, or other, that must be questioned before a system's overall worth can be estimated and that are of primary importance to the

decision authority in reaching a decision to allow the system to advance into the next acquisition phase. (DoD Directive 5000.3).

#### Data Item Description

(AFR800-14)

A form which specifies an item of data required to be furnished by a contractor. This form specifically defines the content, preparation instructions, format and intended use of each data product.  
(AFR 310-1)

#### Decision Analysis

(ROWE)

A methodology of decomposition of the decision-making process into parts, whereby the appropriate data can be associated with the parts, to provide a rational basis for decision making.

#### Decision Making

(ROWE)

A dynamic process of interaction, involving information and judgment among participants who determine a particular policy choice. Decision models are either models of the decision-making process itself, or analytical models (e.g., decision trees, decision matrices) used as aids in arriving at the decisions. Decision theories usually are in relation to the process itself.

#### Decision Matrices

(ROWE)

Matrices whose elements exhibit quantitative relationships (cardinal or ordinal) among sets of factors coming into play in the decision-making process.

#### Decision Tree

(ROWE)

A device used to portray alternative courses of action and relate them to alternative decisions showing all consequences of the decision. The tree represents alternative courses or series of actions related to a previous decision.

#### Decisive Decision Conditions

(ROWE)

Conditions in which the preference between values on a utility scale is clearly discernible because ranges of uncertainty of the two values do not overlap (in the case of uniform distributions of

uncertainty) or are below a certain error level (for normal distributions of uncertainty).

#### Degree of Uncertainty

(ROWE)

That proportion of information about a total system that is unknown in relation to the total information about the system.

#### Delphi Technique

(ROWE)

An iterative method designed to produce a consensus by repeated queries of an individual with feedback of group responses. Members of the group do not interact directly.

#### Descriptive Uncertainty

(ROWE)

The absence of information about the completeness of the description of the degrees of freedom of a system.

#### Designated Approving Authority

(AFR205X)

An official designated to approve the operation of automatic data processing systems at the automatic data processing facilities under his or her jurisdiction for storage of classified or sensitive unclassified information or for critical processing.

#### Deviation

(AFR300-15)

A written authorization, granted prior to the development of a CPCI, to depart from a particular performance or design requirement; a specification for a specific number of units; a specific period of time; or established standards.

#### Documentation

(AFOTECPS)

All of the written work describing operating and maintenance procedures for a system.

#### Documentation Consistency

(AFOTECPS)

A measure of the consistency in the information provided in support system documentation.

## Documentation Descriptiveness

(AFOTEC P5)

A measure of the descriptiveness of the information provided in support system documentation.

## Documentation Modularity

(AFOTEC P5)

A measure of the modular organization of information provided in support system documentation.

## Documentation Simplicity

(AFOTEC P5)

A measure of the ease of use and lack of complexity in the information provided in computer system documentation.

## Embedded Computer Resources

(AFOTEC P1)

Computer resources incorporated as integral parts of, dedicated to, required for direct support of, or for the upgrading or modification of major or less than major system(s). (Excludes ADP resources as defined and administered under AFR 300 series.) (USAF/RD/LE Policy letter, 13 October 1981).

## Embedded Computer System (ECS)

(AFOTEC P1)

a) A computer that is integral to an electromechanical system and that has the following key attributes:

- (1) Physically incorporated into a large system whose primary function is not data processing.
- (2) Integral to, or supportive of, a larger system from a design, procurement, and operations viewpoint.
- (3) Inputs include target data, environmental data, command and control, etc.
- (4) Outputs include target information, flight information, control signals, etc.

b) In general, an embedded computer system (ECS) is developed, acquired, and operated under decentralized management. (DoD Directives 5000.1, 5000.2).

(AFOTEC P5)

A computer that is integral to an electronic or electromechanical system (e.g., aircraft, missile, spacecraft, communications device) from a design, procurement, and operational viewpoint.



Empirical

(ROWE)

Originating in or based on observation or experience.

Equitable Risk

(ROWE)

A risk agent receives direct benefits as a result of exposure to a risk, and the knowledge of the risk is not purposely withheld from the risk agent.

Estimation

(ROWE)

The assignment of probability measures to a postulated future event.

Estimator Uncertainty

(ROWE)

Uncertainty in measurement resulting from deliberate use of less complex measures such as central value estimates of dispersion and smoothing functions for time-dependent parameters.

Evaluation

(ROWE)

Comparison of performance of an activity with the objectives of the activity and assignment of a success measure to that performance.

Evaluation Criteria

(AFOTECPI)

Standards by which achievement of required operational effectiveness/suitability characteristics or resolution of technical or operational issues may be judged. For full-scale development and beyond, evaluation criteria must include quantitative goals (the desired value) and thresholds (the value beyond which the characteristic is unsatisfactory) whenever possible. (DoD Directive 5000.3).

Event

(ROWE)

A particular point in time associated with the beginning or completion of an activity, and possibly accompanied by a statement of the benefit or result attained or to be attained because of the completion of an activity.

### Expandability

(AFOTEC P5)

A measure of the ease with which the functional capability of computer hardware or software may be expanded.

### Expected Value, Use Of

(ROWE)

Valuation of an uncertain numerical event by weighting all possible events by their probability of occurrence and averaging.

### Expert Judgment

(ROWE)

Designating the relevance of opinions of persons well informed in an area for estimates (e.g., forecasts of economic activity).

### Exposure (to risk)

(ROWE)

The condition of being vulnerable to some degree to a particular outcome of an activity, if that outcome occurs.

### Extrapolation/Projection

(ROWE)

The technique of estimating the future by a continuation of past trends without attempts to understand the underlying phenomena.

### Facility

(AFOTEC P5)

The physical plant and the services it provides; specific examples are physical space, electrical power, physical and electromagnetic (TEMPEST) security, environmental control, fire safety provisions, and communications availability.

### Feasible

(ROWE)

That which is possible to do, realistically.

### Feedback

(ROWE)

The return of performance data to a point permitting comparison with objective data, normally for the purpose of improving performance (goal-seeking feedback), but occasionally to modify the objective (goal-changing feedback).

## Firmware

(AFOTECPI)

- a) Computer programs and data loaded in a class of memory that cannot be dynamically modified by the computer during processing.
- b) Hardware that contains a computer program and data that cannot be changed in its application environment.

Note 1. The computer programs and data contained in firmware are classified as software; the circuitry containing the computer program and data is classified as hardware. (Data and Analysis Center for Software).

## Functional Configuration Audit (FCA)

(AFR300-15)

The formal examination of CPCI to verify that the performance specified in the SS has been achieved.

## Independent Verification and Validation (IV&V)

(AFOTECPI)

An independent assessment process structured to ensure that computer programs fulfill the requirements stated in system and subsystem specifications and satisfactorily perform the functions required to meet the user's and supporter's requirements. IV&V consists of three essential elements: independence, verification, and validation:

- (1) Independent. An organization/agency which is separate from the software development activity from a contractual and organizational standpoint.
- (2) Verification. The evaluation to determine whether the products of each step of the computer program development process fulfill all requirements levied by the previous step.
- (3) Validation. The integration, testing, and/or evaluation activities carried out at the system/subsystem level to evaluate the developed computer program against the system specifications and the user's and supporter's requirements. (AFR 88-14)

## Individual Risk Evaluation

(ROWE)

The complex process, conscious or unconscious, whereby an individual accepts a given risk.

### Inequitable Risk

(ROWE)

A risk agent is exposed to a risk and receives no direct benefits from such exposure, or the knowledge of the risk is purposely withheld from him.

### Interdependence

(ROWE)

A property shared by two or more entities whenever the performance of any one affects the performance of some or all the rest.

### Interoperability

(AFOTEC P5)

A measure of the degree to which computer hardware or software can interface to and operate with other similar computer hardware or software.

### Intrinsic Parameter

(ROWE)

A variable whose measurement is based on the value system of an individual and his perception of these values.

### Loss Function

(ROWE)

A function used in decision theory for evaluating the losses incurred when certain decisions are made under uncertainty. If the loss function is independent of the decision value used, it is frequently called a cost function.

### Maintainability

(AFOTEC P3)

Those characteristics of software which affect the ability of the software programmer to correct errors, enhance system capabilities through software changes, and modify the software to be compatible with hardware changes.

(AFOTEC P5)

The probability that a system out of service for maintenance can be properly repaired and returned to service in a stated elapsed time.

### Maintenance Documentation

(AFOTEC P5)

The documentation that describes the maintenance of computer system hardware and software.

Measurable

(ROWE)

a) Capable of being sensed, that which is sensed being convertible to an indication; the indication can be logical, axiological, numerical, or probabilistic. If probabilistic, it is empirical and subjective.

b) Comparable to some unit designated as standard.

Measured Risk Level

(ROWE)

The historic, measured, or modeled risk associated with a given activity.

Measurement Uncertainty

(ROWE)

The absence of information about the specific value of a measurable variable.

Methodology

(ROWE)

An open system of procedures.

Model

(ROWE)

An abstraction of reality that is always an approximation to reality.

Module

(AFR300-15)

A program unit that is discrete and identifiable with respect to compiling and combining with other units.

Nominal Scale (taxonomy)

(ROWE)

A classification of items that can be distinguished from one another by one or more properties.

Objective Function

(ROWE)

A specified mathematical relationship between a dependent variable (e.g., overall measure of benefits) and a set of independent variables (e.g., individual benefit measures and their relative

weights). In choosing among alternatives, the decision maker typically seeks to maximize the (dependent variable of the) objective function.

#### Operational Effectiveness

(AFOTECPI)

The overall degree of mission accomplishment of a system used by representative personnel in the context of the organization, doctrine, tactics, threat (including countermeasures and nuclear threats), and environment in the planned operational employment of the system. (DoD Directive 5000.3)

#### Operational Suitability

(AFOTECPI)

The degree to which a system can be satisfactorily placed in field use, with consideration being given availability, compatibility, transportability, interoperability, reliability, wartime usage rates, maintainability, safety, human factors, manpower supportability, logistic supportability, and training requirements. (DoD Directive 5000.3)

#### Opinion Survey/Sampling

(ROWE)

Any procedure for obtaining by oral or written interrogation or both the views of any portion of the affected population regarding benefit levels expected, their utility, and/or relative importance. Typically, scientific sampling procedures would be used to maximize (for a given level of effort) the accuracy and precision of the results obtained.

#### Opportunity Cost

(ROWE)

The value to society of the next best alternative use of a resource. This is the true economic cost to society of using a resource for a specific purpose or in a specific project.

#### Ordinal Scale (rank scale)

(ROWE)

An ordering (ranking) of items by the degree to which they satisfy some criterion.

#### Paradigm

(ROWE)

A structured set of concepts, definitions, classifications, axioms, and assumptions used in providing a conceptual framework for studying a given problem.

### Parametric Variation

(ROWE)

A technique for sensitivity analysis of any given model in which the values of parameters that are input to the model's calculation are systematically varied to permit observation of how such variation affects the model's output (especially ranking of alternatives).

### Personnel

(AFOTECPS)

A general term for the experience, education, and quantity of people who are assigned to the software support facility either directly or indirectly maintaining the ECS. It includes Management, Technical, Support, and Contractor resources.

### Personnel Profile

(AFOTECPS)

The characteristics that describe the experience, education, and quantity of software support facility personnel.

### Physical Configuration Audit (PCA)

(AFR300-15)

The formal examination of the coded version of a computer program configuration item against its technical documentation.

### Precision

(ROWE)

The exactness with which a quantity is stated, that is, the number of units into which a measurement scale of that quantity may be meaningfully divided. The number of significant digits is a measure of precision.

### Predictive Modeling

(ROWE)

Use of any mathematic model that estimates or predicts the value of a dependent variable in terms of component factors specified as independent variables.

### Preference

(ROWE)

Assignment of rank to items by an agent when the criterion used is utility to the ranking agent.

Probability

(ROWE)

A numerical property attached to an activity or event whereby the likelihood of its future occurrence is expressed or clarified.

Probability Distribution

(ROWE)

The representation of a repeatable stochastic process by a function satisfying the axioms of probability theory.

Probability of Occurrence

(ROWE)

The probability that a particular event will occur, or will occur in a given interval.

Probability Threshold

(ROWE)

A probability of occurrence level for a risk below which a risk agent is no longer concerned with the risk and ignores it in practice (Threshold of concern).

Product Baseline

(AFR300-15)

The initial approved product configuration identification.

Product Verification Review (PVR)

(AFR300-15)

A formal review conducted by the developer for each CPCI at the end of the development phase to establish the Product Baseline for that CPCI and to ensure preparation for the Test Phase has been completed.

Program Manager

(AFR800-14)

The generic term used to denote a single Air Force manager (System Program Director, Program/Project Manager, or System/Item Manager) during any specific phase of the acquisition life cycle. (AFR 800-2).

Program Management Directive (PMD)

(AFR800-14)

The official HQ USAF management directive used to provide direction to the implementing and participating commands and satisfy documentation requirements. It will be used during the entire acquisition



cycle to state requirements and request studies as well as initiate, approve, change, transition, modify or terminate programs. The content of the PMD, including the required HQ USAF review and approval actions, is tailored to the needs of each individual program. (AFR 800-2)

Program Management Plan (PMP)

(AFR800-14)

The document developed and issued by the Program Manager which shows the integrated time-phased tasks and resources required to complete the task specified in the PMD. The PMP is tailored to the needs of each individual program. (AFR 800-2)

Program Office (PO)

(AFR800-14)

The field office organized by the Program Manager to assist him in accomplishing the program tasks. (AFR 800-2)

Program Support Tools

(AFOTTECP3)

General debug aids, test/retest software, trace software/hardware features, use of compiler/link editor, library management/configuration management/text editor/display software tools.

Program Test Plan

(AFOTTECP3)

Set of descriptions and procedures for how the program is to be (or can be, or has been) tested.

Propensity for Risk Acceptance

(ROWE)

An individual, subjective trait designating the degree of risk one is willing to subject himself to for a particular purpose.

Quality Assurance (QA)

(AFR300-15)

All actions that are taken to assure that a development organization delivers products that meet performance requirements and adhere to standards and procedures.

Quantification

(ROWE)

The assignment of a number to an entity or a method for determining a number to be assigned to an entity

## Reliability

(ROWE)

The probability that the system will perform its required functions under given conditions for a specified operating time.

## Residual Risk

(AFR205X)

That portion of risk which remains after security measures have been applied.

## Risk

(AFR205X)

The loss potential which exists as the result of threat/vulnerability pairs. Reducing either the threat or the vulnerability reduces the risk.

(ROWE)

The potential for realization of unwanted, negative consequences of an event.

## Risk Acceptance

(ROWE)

Willingness of an individual, group, or society to accept a specific level of risk to obtain some gain or benefit.

## Risk Acceptance Function

(ROWE)

A subjective operator relating the levels of probability of occurrence and value of a consequence to a level of risk acceptance.

## Risk Acceptance Level

(ROWE)

The acceptable probability of occurrence of a specific consequence value to a given risk agent.

## Risk Acceptance Utility Function

(ROWE)

The profile of the acceptability of the probability of occurrence for all consequences involved in a risk situation for a specific risk agent.

## Risk Agent

(ROWE)

See Valuing Agent.

## Risk Analysis

(AFR205X)

A part of risk management that is used to minimize risk by effectively applying security measures commensurate with the relative threats, vulnerabilities, and values of the resources to be protected. (The value of the resources includes impact on the organizations the automatic data processing system supports, and impact of the loss or unauthorized modification of data). Risk analysis may be thought of as consisting of four modules: sensitivity assessment, risk assessment, economic assessment, and security test and evaluation.

## Risk Assessment

(AFR205X)

A detailed study of the vulnerabilities, threats, likelihood, loss or impact, and theoretical effectiveness of security measures. The results of a risk assessment may be used to develop security requirements and specifications.

(ROWE)

The total process of quantifying a risk and finding an acceptable level of that risk for an individual, group, or society. It involves both risk determination and risk evaluation.

## Risk Averse

(ROWE)

Displaying a propensity against taking risks.

## Risk Aversion

(ROWE)

The act of reducing risk.

## Risk Baseline

(CURRENT)

The risk probability density function and the associated magnitude of consequence for the potential negative outcomes.

## Risk Consequence

(ROWE)

The impact to a risk agent of exposure to a risky event.

## Risk Conversion Factor

(ROWE)

A numerical weight allowing one type of risk to be compared to another type.

Risk Determination

(ROWE)

The process of identifying and estimating the magnitude of risk.

Risk Estimation

(ROWE)

The process of quantification of the probabilities and consequence values for an identified risk.

Risk Evaluation

(ROWE)

The complex process of developing acceptable levels of risk to individuals or society.

Risk Evaluator

(ROWE)

A person, group, or institution that seeks to interpret a valuing agent's risk for a particular purpose.

Risk Identification

(ROWE)

The observation and recognition of new risk parameters, or new relationships among existing risk parameters, or perception of a change in the magnitude of existing risk parameters.

Risk Management

(AFR205X)

The total process of identifying, controlling, and minimizing uncertain events. The process of obtaining and maintaining DAA approval is a major element of the risk management program. The process facilitates the management of automatic data processing (ADP) security risks by each level of ADP management throughout the ADP life cycle. The approval process consists of three elements: risk analysis, certification, and approval.

Risk Profile Baseline

(CURRENT)

The measure of information and/or requirements which serve as the zero reference against which negative (and positive) outcomes can be determined.

Risk Proportionality Derating Factor

(ROWE)

Quantifying the degree to which risks become less acceptable as indirect benefits to the risk agent declines.

Risk Proportionality Factor

(ROWE)

That portion of the total societal risk that society will accept for a new technology.

Risk Reduction

(ROWE)

The action of lowering the probability of occurrence and/or the value of a risk consequence, thereby reducing the magnitude of the risk.

Risk Reference

(ROWE)

Some reference, absolute or relative, against which the acceptability of a similar risk may be measured or related; implies some overall value of risk to society.

Risk Referent

(ROWE)

A specific level of risk deemed acceptable by society or a risk evaluator for a specific risk; it is derived from a risk reference.

Risky Shift

(ROWE)

The tendency of certain groups to become more extreme or take riskier positions in their judgments than they would, acting as individuals.

Sensitivity Analysis

(ROWE)

A method used to examine the operation of a system by measuring the deviation of its nominal behavior due to perturbations in the performance of its components from their nominal values.

Simulation

(AFR800-14)

The representation of physical systems or phenomena by computers, models or other equipment.

## Software

(AFOTECPI)

A set of computer programs, procedures, and associated documentation concerned with the operation of a data processing system.

(CURRENT)

The programs which execute in a computer. The data input, output, controls upon which program execution depends and the documentation which describes, in a textual medium, development and maintenance of the programs.

## Software Error

(CURRENT)

The human decision (inadvertent or by design) which results in the inclusion of a fault in a software product.

## Software Fault

(CURRENT)

The presence or absence of that part of a software product which can result in software failure.

## Software Maintainability

(AFOTECPI)

The ease with which software can be changed in order to:

- (1) Correct errors.
- (2) Add or modify system capabilities through software changes.
- (3) Delete features from programs.
- (4) Modify software to be compatible with hardware changes.

(CURRENT)

A quality of software which reflects the effort required to perform software maintenance actions.

## Software Maintenance

(CURRENT)

Those actions required for:

- (1) Correction. Removal, correction of software faults
- (2) Enhancement. Addition/deletion of features from the software
- (3) Conversion. Modification of the software because of environment (data hardware) changes.

## Software Maintenance Environment

(CURRENT)

An integration of personnel support systems and physical facilities for the purpose of maintaining software products.

## Software Maintenance Measures

(CURRENT)

Measures of software maintainability and environment capabilities to support software maintenance activity.

## Software Management

(CURRENT)

The policy, methodology, procedures, and guidelines applied in a software environment to the software development/maintenance activities. Also, those personnel with software management responsibilities.

## Software Reliability

(CURRENT)

A quality of software which reflects the probability of failure free operation of a software component or system in a specified environment for a specified time.

## Software Portability

(CURRENT)

A quality of software which reflects the effort required to transfer the software from one environment (hardware and system software) to another.

## Software Support Facility (SSF)

(AFOTEC P5)

The Facility which houses and provides services for the Support Systems and Personnel required to maintain the software for a specific ECS.

## Software Supportability

(CURRENT)

A measure of the adequacy of personnel, resources, and procedures to facilitate:

- (1) Modifying and installing software
- (2) Establishing an operational software baseline
- (3) Meeting user requirements.

## Specification

(AFR300-15)

A document that describes the requirements for the development or acquisition of ADPE and/or software.

## Standards

(AFOTTECP3)

Procedures, rules, and conventions used for prescribing disciplined program design and implementation.

## States of Nature

(ROWE)

A concept from decision theory. In decision making under uncertainty, the outcomes (numerical results) associated with each available alternative are considered to be predictable as a set of  $n$  discrete values depending on conditions beyond the decision maker's control and for which he has no useful estimates of the respective probabilities. The  $n$  sets of conditions under which each one of the outcomes is expected are termed "states of nature."

## Structured Value (structured value analysis)

(ROWE)

The resultant value of a particular value set evaluated for a particular data set. This value lies between zero and unity and allows many data sets to be ranked numerically to relation to one another.

## Structured Value Analysis

(ROWE)

A multistage procedure for assessing the value of an action, project alternative, and so on, incorporating individual techniques at each stage for computing from quantitative measures of individual components a single figure expressing the overall value. A multistage procedure for assessing the value of an action, project, alternative, and so on, by structuring the complete entity into component elements, to each of which a numeric measure of value (positive or negative) can be assigned. These are then converted to a common utility scale. Each component is assigned a weight expressing its relative significance in determining overall value of the entity. A single figure of worth or value is then computed from measures and weights of all individual components. The procedure permits considerable flexibility in choice of techniques used to perform each necessary optimal step.



### Subjective Probabilities

(ROWE)

The assignment of subjective weights to possible outcomes of an uncertain event where weights assigned satisfy axioms of probability theory.

### Support Personnel

(AFOTECPS)

A general term for military or DoD civilian personnel whose skills are necessary for the software support facility to function but who do not directly support ECS software maintenance.

### Support System

(AFOTECPS)

Any automated system used to change, test, or manage the configuration of ECS software and associated documentation. Includes but is not limited to Host Processor, Software Bench, Laboratory-Integrated Test Facility, Operational-Integrated Test Facility, and Configuration Management System.

### Support System Facility

(AFOTECPS)

The facility resources that must be available for the software support resources to accomplish a specific task(s) (see General Facility).

### Surrogate or Proxy Measures

(ROWE)

The use of a related quantity as a proxy for an unknown or difficult-to-measure value. The relationship may be established by arm-chair analysis, correlation techniques, scientific studies, or other means.

### System

(ROWE)

- a) A complex entity formed of many, often diverse, parts subject to a common plan or serving a common purpose.
- b) A composite of equipment, skills, and techniques capable of performing and/or supporting an operation.

### System Design Review (SDR)

(AFR300-15)

A formal review of the system design approach for an ADS.

### System Requirements Review (SRR)

(AFR300-15)

A formal review of the requirements for an ADS.

### System Software

(AFOTECP5)

All of the software that is part of the software support facility computer system. It is never or seldom accessed directly by software support facility personnel; it controls the processing of application software. It includes the Operating System, Source Code Editor, Language Translator, Link Editor/Loader, Librarian/File Manager, Data Base Manager, and Automated Software Development Tool.

### Taxonomy

(ROWE)

The identification and definition of properties of elements of the universe; a disaggregation, as contrasted with systematics (which is an aggregation) and as contrasted with morphology (which encompasses both taxonomy and systematics).

### Test Analysis Report (RT)

(AFR300-15)

A document containing the results and analyses of tests executed during the Test Phase.

### Threshold

(ROWE)

A discontinuous change of state of a parameter as its measure increases. One condition exists below the discontinuity, and a different one above it.

### Transfer

(AFR800-14)

That point in time when the designated Supporting Command accepts program management responsibilities from the Implementing Command. This includes logistic support and related engineering and procurement responsibilities. (AFR 800-4)

### Turnover

(AFR800-14)

That point in time when the operating command formally accepts responsibility from the Implementing Command for the operation and maintenance of the system, equipment, or computer program acquired. (AFR 800-19)

Uncertainty

(ROWE)

The absence of information; that which is unknown.

User

(AFR205X)

Any persons (or organizations) having access to an automatic data processing system via communication through a remote device or who is allowed to submit input to the system through other media (for example, tape or card decks). (Does not include those persons or organizations defined as customers.)

Valuation

(ROWE)

The act of mapping an ordinal scale onto an interval scale (i.e., assigning a numerical measure to each ranked item based on its relative distance from the end points of the interval scale... assigning an interval scale value to a risk consequence.

Value

(ROWE)

A quality quantified on a scale expressing the satisfaction of man's intrinsic wants and desires.

Value Function (structured value analysis)

(ROWE)

A function relating points on the parameter measurement scale to the value scale for a particular parameter. These functions may result from explicit information or may be arrived at through value judgment.

Value Set (structured value analysis)

(ROWE)

A specific set of model parameters made up of terms and factors, expressed in particular measurement scales, value functions, and weights.

Valuing

(ROWE)

The act of assigning a value to a risk consequence.

Valuing Agent

(ROWE)

A person or group of persons who evaluates directly the consequence of a risk to which he is subjected. A risk agent.

Verification/Validation (of computer programs)

(AFR800-14)

The process of determining that the computer program was developed in accordance with the stated specification and satisfactorily performs, in the mission environment, the function(s) for which it was designed.

Weight (structured value analysis)

(ROWE)

The relative importance of terms in a model expressed as a decimal fraction; weights for a set of terms add to unity.

Weighting Factor

(ROWE)

A coefficient used to adjust variable accuracy to a subjective evaluation; these factors are usually determined through surveys, Delphi sessions, or other formats of expressing social priorities.

**Appendix C**  
**DoD/Air Force Management Policy**

APPENDIX C  
POLICY DIRECTIVES

C.1 GENERAL.

This appendix summarizes and annotates the directives of higher authorities and the military services. Material in this appendix is derived from reference 5.35.

These directives were identified by reviewing the system acquisition and management directives of OMB, DoD, and the Air Force, and noting all references which deal specifically with risk analysis. Some references imply the need for risk analysis, but do not explicitly state such requirement. Although additional documents were reviewed, only those listed were found to contain material relevant to risk analysis.

C.2 HIGHER LEVEL REQUIREMENTS.

This section lists excerpts and comments briefly on OMB and DoD policies and directives relating to risk assessment.

C.2.1 Office of Management and Budget.

OMB Circular A-109. Major System Acquisition (5 April 1976).

Paragraph 7. "Each agency acquiring major systems should... tailor an acquisition strategy for each program. ...The strategy should typically include...methods for analyzing and evaluating contractor and Government risks."

C.2.2 Department of Defense.

DoD Directive (DoDD) 5000.1. Major System Acquisition (29 March 1982).

Paragraph C.2.C.(3). To achieve program stability, DoD components will "estimate and budget realistically, and fund adequately, procurement

(research, development, and production), logistics and manpower for major systems."

Paragraph E.4.C.(1)(a). This paragraph states that it may be reasonable to delay Milestone II decisions until some development efforts are accomplished in order to "reduce risk and uncertainty before the commitment to a major increase in the application of resources toward full-scale development is made."

Paragraph E.8. "Commensurate with risk, such approaches (to reduce acquisition time) as developing separate alternatives in high-risk areas, should be encouraged."

DoDI 5000.2 Major Systems Acquisition Procedures (March 8, 1983).

No references to risk analysis appear in the body of the text, however, paragraphs D.3.e.(1)(a) and D.3.e.(2)(a) refer to the need for System Concept Papers (SCP's) and Decision Coordinating Papers (DCP's) to establish and identify goals, thresholds, and threshold ranges (emphasis supplied), thus recognizing the concept of risk.

Enclosure (4) Format for SCP and DCP.

"VIII. Technological Risks of Selected Alternative. For Milestone I (SCP), identify key areas of technological risk which must be reduced by R&D and validated by T&E before Milestone II. For Milestone II (DCP), discuss T&E results that show all significant risk areas have been resolved. Also, for Milestone II, verify that technology is in hand and also engineering (rather than experimental) effort remains."

DoDI 5000.38. Production Readiness Reviews (24 January 1979).

Paragraph A.2. "The objective of a Program Readiness Review (PRR) is to verify that the production, design, planning, and associated preparations for a system have progressed to the point where a production commitment can be made without incurring unacceptable risks of breaching thresholds of schedule, performance, cost, or other established criteria."

Paragraph E.4. "The DPESO (DoD Product Engineering Services Office) independent production readiness assessment will consist of objective conclusions based on the findings of the PRR and other investigations."

This assessment will identify potential problem areas which constitute production, cost, or schedule risks. Each risk will be expressed in terms of its relative magnitude and potential consequences. (Emphasis supplied.)

DoDI 7041.3. Economic Analyses and Program Evaluation for Resource Management (October 19, 1972).

Enclosure (2)

Paragraph B.7. "Risk/Uncertainty Analysis. Risk assessments will be made to determine the expectation or probability that program/project objectives will be realized by following a specific course of action with constraints of time, cost, and technical performance. (Emphasis supplied.) Actual costs and outputs of many DoD projects differ from those expected at the time of decision. For those cases, and in particular for major weapon systems covered by a Selected Acquisition Review Report or subject to review by the Defense System Acquisition Review Committee (DSARC), the impact which could result from this variability should be evaluated."

Paragraph B.7.a. "Independent parametric cost estimates can provide an early test of the reasonableness of cost estimates. Independent parametric cost estimates will be made at key decision points for major weapon systems, e.g., during concept formulation and prior to making major commitments of funds for development and production. These estimates generally consider cost at high levels of aggregation and are predicated on actual historical costs encountered in like or similar programs. As such, they incorporate costs for expected uncertainties on the average. (1) Costs should be derived by parametric techniques and expressed as feasible ranges in terms of the parameters which drive them. It is most important that estimates be presented as cost ranges related to the probable values of system parameters, characteristics, or attributes which are determined by costs. (Emphasis supplied). (2) These estimates will be available for each DSARC review. Parametric estimates will be derived independent of functional, program manager or contractor influence. (3) When the independent parametric cost estimate



differs from the program manager's current estimate, the latter estimate will be used for economic analysis/program evaluations. Once a program estimate is established as a baseline, a program/project manager will manage his program within that limitation. (4) The program manager's current estimate will be an assessment of the ultimate cost expected for a program/project including undefinitized contingencies. (Emphasis supplied.) As such, the program manager's current estimate should be relatively stable over long periods of time and not change with small incremental changes to the approved program, funding changes, or financial fluctuations. To the extent possible, schedules, and funding should be structured to accommodate program uncertainties and unforeseen problems." (Emphasis supplied.)

Paragraph 8.7.b. "Special degrees of risk/uncertainty associated with a particular program/project, may be pointed out quantitatively in an analysis and used for program review purposes. Probability estimates can be developed by testing the sensitivity of key variables on estimated costs and performance. The probability that each of the possible cost or output estimates may be realized should be discussed narratively when there is no basis for a quantitative estimate." (Emphasis supplied.)

Paragraph 8.7.c. Estimates will be expressed in terms of performance thresholds, goals, or ranges. Program/project estimates will include the limits within which ultimate program cost and technical performance is expected to fall."

### C.3 SERVICE REQUIREMENTS (U.S. AIR FORCE).

The following Air Force directives address consideration of program risk as revealed by excerpt or editorial summation.

#### C.3.1 Air Force Regulation AFR 173-11.

Independent Cost Analysis Program (12 Dec 1980).

Paragraph 6. Definition and Scope of the Independent Cost Analysis (ICA).

Paragraph 6.j. "Will contain a detailed risk assessment to include risk related to the cost estimating techniques employed and with technical and schedule uncertainties that may have an impact on cost estimates. It will also include sensitivity analyses of critical assumptions and cost driving parameters."

Paragraph 7.e. "For cost elements with a high degree of uncertainty, the ICA will provide sensitivity analysis using frequency distributions or ranges of cost. The probability distributions used to prepare range estimates, as well as the proper assumptions, must be provided. "Prediction intervals around cost estimating relationships (CERs) or Monte Carlo simulations will be used as proper in quantifying risk." (Emphasis supplied.)

Paragraph 9.d. "The ISR will address the potential risk in the program office estimate by identifying 'risk' areas and their probable and possible cost impact." (ISR means independent schedule review.)

#### C.3.2 AFR 70-15. Source Selection Policy and Procedures.

Paragraph 1-4.d. "The source selection process shall focus adequate attention on the program risk and uncertainties during solicitation, proposed evaluation, and selection phases.

- a) Offerors should not be penalized for the identification of risk associated with their proposals. Proposals should be credited when realistic approaches for risk resolution are provided.
- b) The procuring activity shall prepare an independent risk assessment before receipt of proposals, to facilitate risk analysis evaluation.

Paragraph 2-2.c(3). "It (the evaluation criteria) must address those high risks and technical uncertainties, which were identified by the offerors and the Government as 'known-unknowns' during the conceptual phase. An indication should also be provided of the relative importance of each criterion for later use in the solicitation."

Paragraph 2-4.d. "...Risk analysis is a part of the evaluation process, and risk assessment for each proposal must be included in all reports to the Source Selection Advisory Council (SSAC) and Source Selection Authority (SSA). Technical risk, as pertains to each proposal, should be rated based on the offeror's risk assessment and the credibility of his proposed approach for eliminating or avoiding such risks."

Paragraph 3-2.a.(2). "The solicitation should...include a discussion of known or potential risks, where there is reason to believe that the potential offerors are not aware of the risks."

Paragraph 3-7.e. "The offerors must be required to submit a risk analysis as part of their proposal which also identifies risk areas and which furnishes an insight to the evaluator as to how the offeror intends to resolve these risks and the alternatives to overcoming the high risk approaches. In order to aid the evaluator in performing the risk analysis, the procuring activity should prepare an independent risk assessment prior to receipt of proposals."

Paragraph 3-8.b.(5). This paragraph states that the SSA must determine cost/price risk inherent in each proposal.

#### Attachment 4 - VIII. Risk Analysis.

This paragraph lists risk analysis documentation format.

#### C.3.3 AFR 800-3. Engineering for Defense Systems, (17 June 1977).

Paragraph 4.b. (In the validation phase)"...certain technical aspects may need to be intensified, such as technical and cost risk reduction, obtaining a best mix of technical requirements, and other considerations or thresholds as may be described in the PMD."

Paragraph 6.f. The AFSC "programs their research and development (R&D) projects to develop and improve systems engineering methods and techniques (system cost effectiveness, risk assessment, technical performance measurement, etc.)."

C.3.4 AFR 800-8. ILS Program, (7 February 1980).

Paragraph 5.r. "Risk analysis and assessment and tradeoff analyses will be conducted, using the latest data available."

C.3.5 AFR 800-9. Manufacturing Management for Air Force Acquisitions (1 October 1979).

Paragraph 2.c. "In the manufacturing assessment of system and design alternatives the program manager will: (1) consider the relationship between several factors (such as producibility, manufacturing risks, productivity, ...) and evaluate their impact on the minimum essential performance requirements."

Attachment 1. Extracts from DoDD 5000.34, 31 October 1977.

Paragraph D.5. "...Production risks, which should be identified as early as possible in the acquisition cycle, shall be reduced to acceptable levels prior to production decision."

C.3.6 Aeronautical Systems Division Regulation (ASDR) 173-1 Aeronautical Systems Division Cost Analysis Program (21 October 1981).

Attachment 5: Cost Estimate Risk Assessment Guidelines.

Paragraph 1. "...The purpose of the risk analysis described below is to alert decision makers to:

- a) Inputs or assumptions where a percent change in an input or assumption value would make at least one half that percent change in the total estimate.
- b) Areas of uncertainty at the time the estimate was prepared...

Paragraph 2. "Generally, risk assessments must be prepared so that while not all possible areas of risk are addressed, the overall amount of

risk of cost growth can be addressed by review of the several highest risk areas identified and discussed."

Paragraph 3. "The risks of cost increase over the estimates to be addressed will be primarily those associated with estimating methods are (sic) available data/information limitations. The risks of strikes, major test or technical approach failures, directed program changes, etc., are not to be addressed."

END

DATE

FILMED

5-88

DTIC